Design • Build • Program • Learn • Compete

SERVO

SERVO

FOR THE ROBOT INNOVATOR

www.servomagazine.com

MAGAZINE

April 2013

WEAPONIZE YOUR WEDGIE

No Background Check Required!

- **Autonomousize Your Quadcoptor**
  Add an APM 2.5+ autopilot to your Kronos Flyer.
- **What's Next From NASA?**
  A lunar mining robot to process water and fuel from lunar soil.

New Mobile App! SERVO Goes Where You Go
Available For:
iOS - Android
Kindle Fire

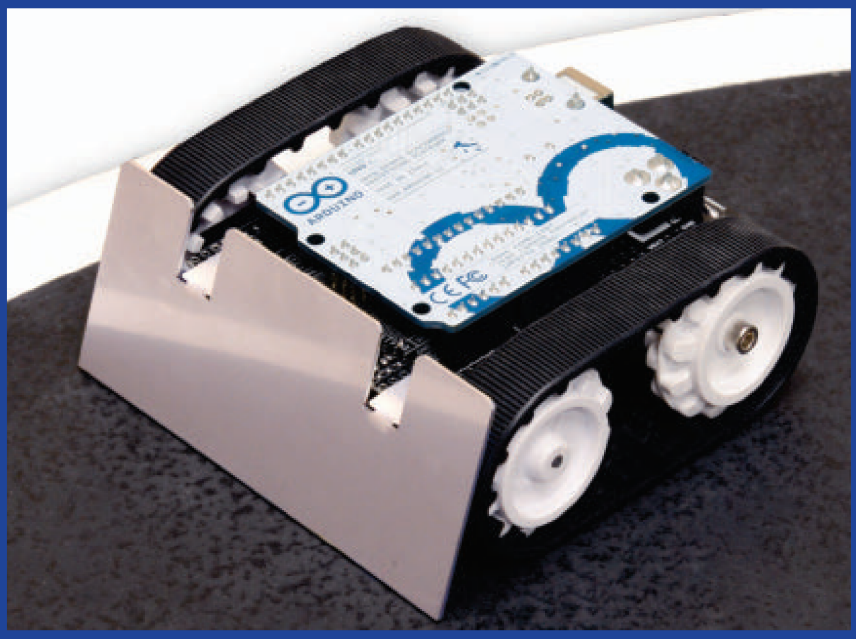
Smart. Powerful. Sleek. Professional.

Four Great Reasons to Choose the *H4* Multi Charger.

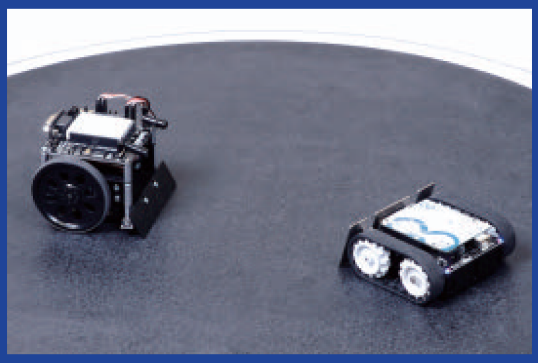
Our high-performance **H4** is a true professional-grade, four-channel, balance charger for all batteries. Designed with four 120-watt power ports, the **H4** features our exceptional Channel Bridge function, allowing you to combine two channels into a single, 240-watt conduit. Its durable and sleek full aluminum case, 32-bit microprocessor and comprehensive charge and discharge capabilities make the **H4** Multi Charger a great fit for discriminating customers everywhere.

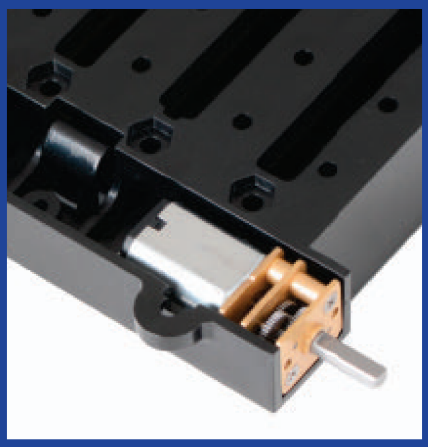
Demand More, Demand HiTEC

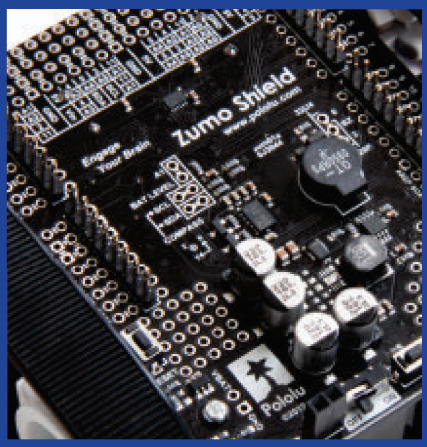
12115 Paine Street • Poway, CA 92064 • 858.748.6948 • www.hitecrcd.com



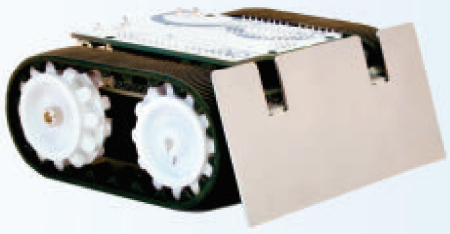
Build and customize your Zumo!







Small enough for Mini-Sumo; flexible enough to make it your own.



Put your Arduino or compatible controller on the right tracks with the Zumo chassis and Arduino shield! The Zumo is a small, tracked robot platform that works with a variety of micro metal gearmotors to allow for a customizable combination of torque and speed. Add a Zumo shield, which includes a dual motor driver, buzzer, and three-axis accelerometer and compass, to make an Arduino-controlled robot that can really throw its weight around!

Pololu
Robotics & Electronics

Find out more at **www.pololu.com/zumo**

SERVO MAGAZINE

04.2013
VOL. 11 NO. 4

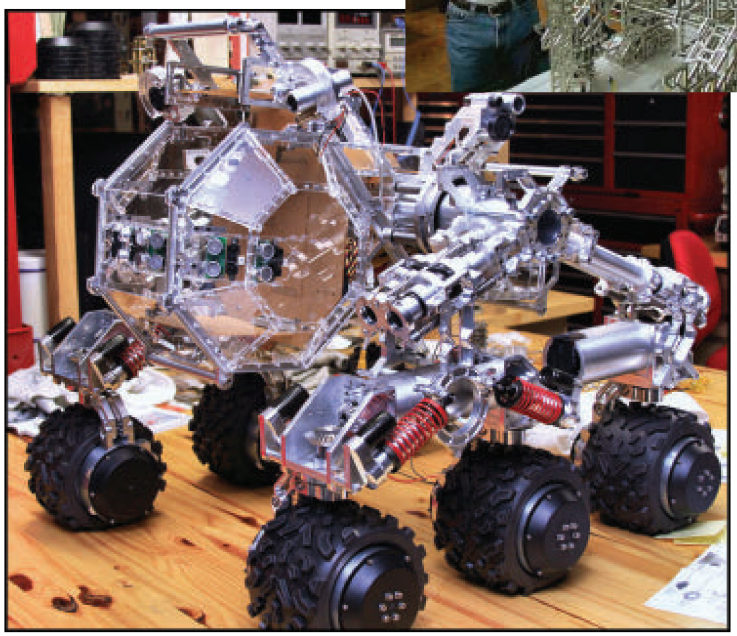
Columns

08 Robytes
by Jeff Eckert
Stimulating Robot Tidbits
- Diggin' It at NASA
- Literally for the Birds
- A UAV to Track Taliban
- And a UAV to Track You
- Creepy Times Two

11 Ask Mr. Roboto
by Dennis Clark
Your Problems Solved Here
Basic Subsumption Architecture

74 Then and Now
by Tom Carroll
Get That Robot Built the Easy Way
Parts to build great robots are all around us. Just use your imagination the next time you're shopping at the local hobby, hardware, marine, RV, or toy store.

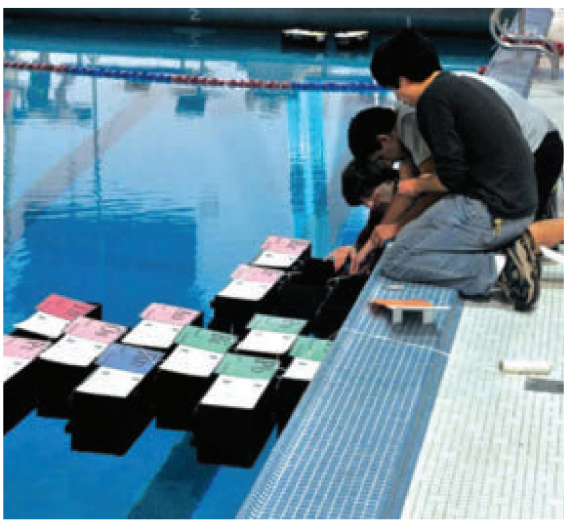
PAGE 74



Departments

06 Mind/Iron
16 Events Calendar
17 Showcase
18 New Products
68 SERVO Webstore
80 Robo-Links
80 Advertiser's Index

20 Bots in Brief
- Black Hornet Gets Green Light
- Moth Balls
- Sad Rats
- Senseless Juggling
- Ro, Ro, Robo Boats
- Bird's Eye View
- STAR Power
- Gone Batty
- Pheupard is the Cat's Meow
- Inside(s) Look



SERVO Magazine **(ISSN 1546-0592/CDN Pub Agree#40702530)** is published monthly for $24.95 per year by T & L Publications, Inc., 430 Princeland Court, Corona, CA 92879. PERIODICALS POSTAGE PAID AT CORONA, CA AND AT ADDITIONAL ENTRY MAILING OFFICES. POSTMASTER: Send address changes to **SERVO Magazine, P.O. Box 15277, North Hollywood, CA 91615** or Station A, P.O. Box 54, Windsor ON N9A 6J5; cpcreturns@servomagazine.com

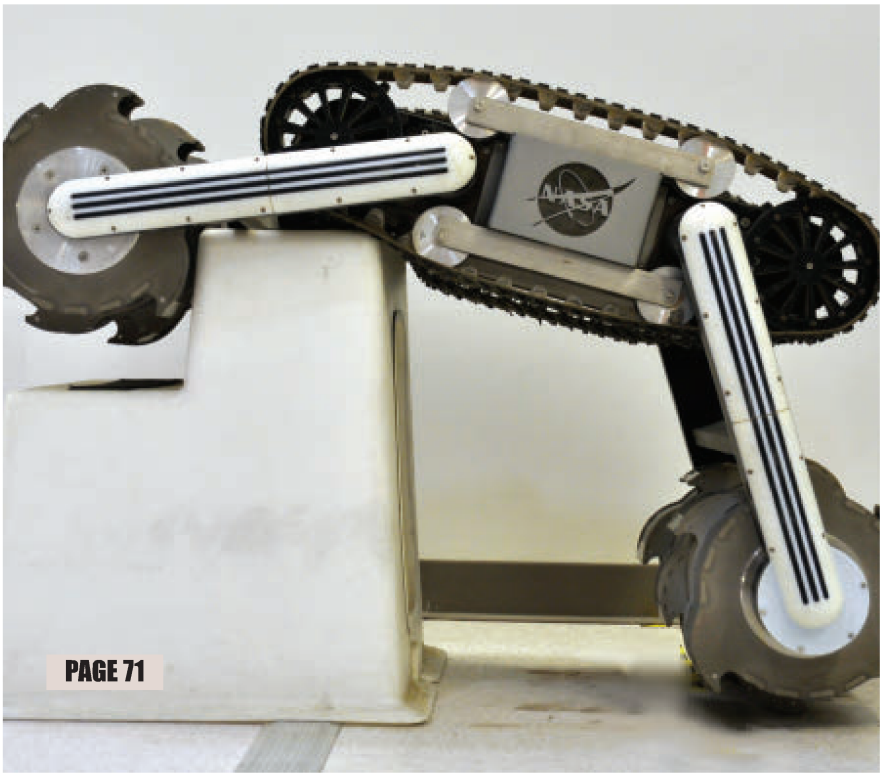
In This Issue ...

36 Build the Kronos Flyer
by Michael Simpson
Part 6: APM 2.5
Per reader requests, this final installment of our quadcopter build will discuss the APM 2.5 controller in detail.

46 Robotic Sensors
by John Blankenship and Samuel Mishal
Part 2: Their Use With Robotic Arms. Using the principles of identification discussed last month, we'll have our mobile robot find and retrieve objects with an arm and gripper.

54 Give Yourself a Weapon!
by Zachary Lytle
As a follow-up to the hugely popular *Give Yourself a Wedgie* project that appeared in the October 2012 issue, this article will show you how to add either a clamp or lifter weapon to your robot.

71 NASA Engineers Building Lunar Mining Robot
by Kevin Berry
Although RASSOR is far from being space-ready, this autonomous machine will ultimately play a key role in being able to process water and fuel from lunar soil.



The Combat Zone...

26 BUILD REPORT:
Mini Bot Hockey Bots — Part 3

29 Is This the Future of Combat Robotics?

30 PARTS IS PARTS:
Product Review:
FingerTech Robotics Power Switch

32 EVENT REPORT:
Robots, Birthday Cake, and Off-Road Trucks!

Mind / Iron

by Bryan Bergeron, Editor

Ticklish Robots

By now, I expect that you're familiar with the android David in the sci-fi film *Prometheus*. David exemplifies near perfection — the ability to speak and understand language, an apparently perfect humanoid body, the ability to effectively lie and deceive, and a sense of self-preservation. What the film failed to reveal, however, was whether David was ticklish.

I focus on this seemingly insignificant ability because it's something that most humans demonstrate and because it seems more easily achieved than, say, the ability to deceive others. Think about it — what could be so difficult? You'd need a sensor or two, and the ability to determine if a body part was touching or being touched by someone or something else.

I started an experiment with a 5DOF robot arm, a few pressure sensors, and an Arduino, and quickly discovered that determining who or what is doing the touching is non-trivial.

For example, to determine whether a pressure sensor response is due to movement of the arm or of something external touching the arm, you have to keep a history of the arm movement. If the arm has been sitting idle for 20 seconds, then you can probably conclude that the pressure sensor reacted to an external force. If the arm was moving, then it could have been either arm or external movement, or both. So, you'd need an optical or IR sensor to determine if someone or something was near.

There's no $39 "Tickle Me Elmo" solution to creating a realistically ticklish android or black box robot. The skin sensors would need to distinguish a light soft touch from, say, a punch or scratch. Then, there's the higher-level processing required to determine whether the soft touch should result in a tickle response, pulling away in a defensive posture, or a slap in the face.

Local sensor processing isn't enough. People often start laughing before being tickled. That is, the anticipation is enough to evoke laughter. Not only would you need to detect, say, a hand sneaking up, approaching your armpit or other sensitive spot, but you'd have to analyze the facial expression of the suspected tickler. A negative expression might suggest a mugger, and not someone to be taken lightly.

As impressive as Watson is playing Jeopardy, I'd be more impressed if it were ticklish. I think that the Touring Test will be passed by computers long before the 'ticklish test.' A reasonable question is "So what?" As David said in *Prometheus*, we humans are most comfortable around our own kind. An android that can't display the full range of human interactions — including being ticklish (and, as an aside, being able to tickle others) — would hardly pass for human.

From a practical perspective, future servant robots that attend to the aging and sick will need a full range of human-like emotions in order to bond with their patients and owners. As far as my experiments go, I've given up on the Arduino as an experimental platform, simply because it doesn't have the processing power. Instead, I'm working with the Parallax Propeller and the Raspberry Pi — both powerful, inexpensive microcontrollers that may have enough speed and memory to monitor multiple sensors and at least guess about the context. Time will tell.

If you've worked in this area, please consider sharing your experience with your fellow readers. SV

FOR THE ROBOT INNOVATOR SERVO MAGAZINE

Published Monthly By
T & L Publications, Inc.
430 Princeland Ct., Corona, CA 92879-1300
(951) 371-8497
FAX **(951) 371-3052**
Webstore Only **1-800-783-4624**
www.servomagazine.com

Subscriptions
Toll Free **1-877-525-2539**
Outside US **1-818-487-4545**
P.O. Box 15277, N. Hollywood, CA 91615

PUBLISHER
Larry Lemieux
publisher@servomagazine.com

ASSOCIATE PUBLISHER/ VP OF SALES/MARKETING
Robin Lemieux
display@servomagazine.com

EDITOR
Bryan Bergeron
techedit-servo@yahoo.com

CONTRIBUTING EDITORS
Jeff Eckert, Tom Carroll, Dennis Clark, Michael Simpson, John Blankenship, Pete Smith, Andrea Suarez, Jenn Eckert, Kevin Berry, R. Steven Rainwater, Zachary Lytle, Samuel Mishal, Mike Jeffries

CIRCULATION DEPARTMENT
subscribe@servomagazine.com

MARKETING COORDINATOR WEBSTORE
Brian Kirkpatrick
sales@servomagazine.com

WEB CONTENT
Michael Kaudze
website@servomagazine.com

ADMINISTRATIVE ASSISTANT
Debbie Stauffacher

PRODUCTION
Sean Lemieux

Copyright 2013 by
T & L Publications, Inc.
All Rights Reserved

All advertising is subject to publisher's approval. We are not responsible for mistakes, misprints, or typographical errors. *SERVO Magazine* assumes no responsibility for the availability or condition of advertised items or for the honesty of the advertiser. The publisher makes no claims for the legality of any item advertised in *SERVO*. This is the sole responsibility of the advertiser. Advertisers and their agencies agree to indemnify and protect the publisher from any and all claims, action, or expense arising from advertising placed in *SERVO*. Please send all editorial correspondence, UPS, overnight mail, and artwork to: **430 Princeland Court, Corona, CA 92879.**

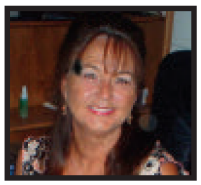
Printed in the USA on SFI & FSC stock.

Solutions^3

www.solutions-cubed.com
designservices@solutions-cubed.com
530.891.8045

Solutions Cubed... hardware made easy

for a 5% discount use coupon code SERVOMAG13 (through 1/1/14)



Robytes

by Jeff and Jenn Eckert

Diggin' It at NASA

The standard formula for a NASA robot is big, intricate, and loaded with more scientific instruments than the average physics lab. The agency (**www.nasa.gov**), however, has also been developing a prototype diggerbot that's such a primordial worker that you could almost have a beer and talk sports with it. The machine — called the Regolith Advanced Surface Systems Operations Robot (RASSOR, pronounced "razor") — is basically designed to toil for years on end with its twin diggers, scraping through layers of lunar soil.

RASSOR accomplishes this via counter-rotating bucket drums that give one end enough traction to stay in place while the other end digs about 30 cm into the soil. Its main mission would be to pull water and ice out of the dirt, and transform the mined chemicals into usable amounts of rocket fuel or breathing air for astronauts. This would require it to operate 16 hours per day for five years. With the drums positioned above the bot's main body, it stands about 2.5 ft tall.

A mission-ready version would weigh about 100 lb. According to the RASSOR team, a new improved RASSOR II should be ready for testing early next year. (See the article in this issue by Kevin Berry that talks about RASSOR in more detail.)

RASSOR climbs a hill in a test at Kennedy Space Center.
Photo credit: NASA.

Discuss this article in the *SERVO Magazine* forums at http://forum.servomagazine.com.

Korean anti-bird robotic ground vehicle protects airports.

Literally for the Birds

It's no secret that animals straying into the path of airplanes constitute a serious problem. Just ask US Airways Capt. "Sully" Sullenberger, who was forced into the Hudson River by a flock of geese. According to an FAA study, reported strikes (97 percent of which involved birds) increased from 1,804 in 1990 to 10,083 in 2011.

During that period, the related costs in civil aviation alone came to an estimated $481 million. So, maybe a new autonomous anti-bird vehicle developed by the Korean Atomic Energy Group and LIG Nex1 (an LG subsidiary) isn't as loopy as it admittedly looks.

The six-wheeler is almost the size of a small car and weighs in at 1.2 tons. It's bird scaring equipment includes a sound system that emits bird predator noises at 100 dB, built-in cameras that can track a 12 inch bird at up to 330 yd, and green lasers designed to scare the flying nuisances away.

At present, the system is being deployed only at Korean airfields, but the company reports considerable interest from international customers.

A UAV to Track Taliban

The PD-100 Black Hornet from Norway's Prox Dynamics.

It may not look a great deal more sophisticated than my S102G toy chopper, but the PD-100 Black Hornet Personal Reconnaissance System from Prox Dynamics (**www.proxdynamics.com**) was developed for serious employment by law enforcement, search-and-rescue personnel, and other government users. It is covered by Norwegian Arms Control provisions and will be sold "only to government institutions and organizations," so there's no point in drooling. In fact, it has recently been adopted by British forces as their latest anti-Taliban tool.

The 8 in long, 15 g molded plastic drone carries three cameras, and soldiers can pilot it manually or program it to fly according to battlefield coordinates using GPS. According to the Brits' Major Adam Foden, "Black Hornet is a game-changing piece of kit. Previously, we would have sent soldiers forward to see if there were any enemy fighters hiding inside a set of buildings. Now, we are deploying Black Hornet to look inside compounds and to clear a route through enemy-held spaces. It has worked very well, and the pictures it delivers back to the monitor are really clear. And Black Hornet is so small and quiet that the locals can't see or hear it."

Few specs were provided, but reports have it that the nanocopter provides up to 30 min of flight time and a top speed of 20 mph. A complete system includes three PD-100 units, a controller with LCD display, and a transport case with charger. It is billed as "affordable," but in the military world, that could mean anything. Presumably, it will not be in the F-16 range.

And a UAV to Track You

If you are a typical Facebook user, you no doubt keep your friends and relatives up to date about your fascinating life with a stream of information about your pet's health, what you had for breakfast at Denny's, how late you worked last night, and so on. You may even supplement that with plenty of photos in case the folks don't actually know what a Grand Slam breakfast looks like. Do you ever get the feeling, though, that some people just aren't getting enough details? If so, then you need MeCam — a microcopter designed exclusively to follow you around all day and take streaming videos of your every move.

Developed by San Francisco based Always Innovating (**www.alwaysinnovating.com**), the quad-rotor UAV launches from your hand and navigates either by voice command or a "follow-me" mode. It then streams the video to an Android or iOS phone or tablet, so you can share it via Facebook, YouTube, or another social networking platform. The good news is that the targeted MSRP is only $50. The bad news is that the company isn't actually building any of them; it is only offering licenses to other entities to produce and sell it. We therefore can't be sure that it will ever really be available, but a company press release predicts that the product could "hit the shelves by the beginning of 2014." We'll see.

The MeCam video 'copter allows you to point and shoot yourself.

Firgelli
www.firgelli.com
LINEAR SERVOS
L12-R Linear Servo
• Direct replacement for regular rotary servos
• Standard 3 wire connectors
• Compatible with most R/C receivers
• 1-2ms PWM control signal, 6v power
• 1", 2" and 4" strokes
• 3-10 lbs. force range
• 1/4" to 1" per second speed ranges
• Compatible with VEX
L16 Linear Actuators
New!
• 2", 4" and 6" strokes
• 10 - 40 lbs. force range
• 1/2" to 1" per second speed ranges
• Options include Limit Switches and Position Feedback
PQ12 Linear Actuator
• Miniature Linear Motion Devices
• 6 or 12 volts, 3/4" stroke
• Up to 5 lbs. force
• Integrated position feedback or limit switches at end of stroke
• External position control available
Linear Actuator Controller (LAC)
• Will drive any Linear Actuator with position feedback
• Up to 24v and 4 Amps
• USB connectivity to drive the actuaor with your computer
• Adjustable speed, limits and sensitivity
L12-NXT Linear Servo
• Designed for LEGO Mindstorms NXT®
• Plugs directly into your NXT Brick
• NXT-G Block available for download
• Can be used with Technic and PF
• Max. speed: 1/2" per sec.
• Pushes up to 5 lbs.
• 2" and 4" strokes
Available Now @
www.firgelli.com

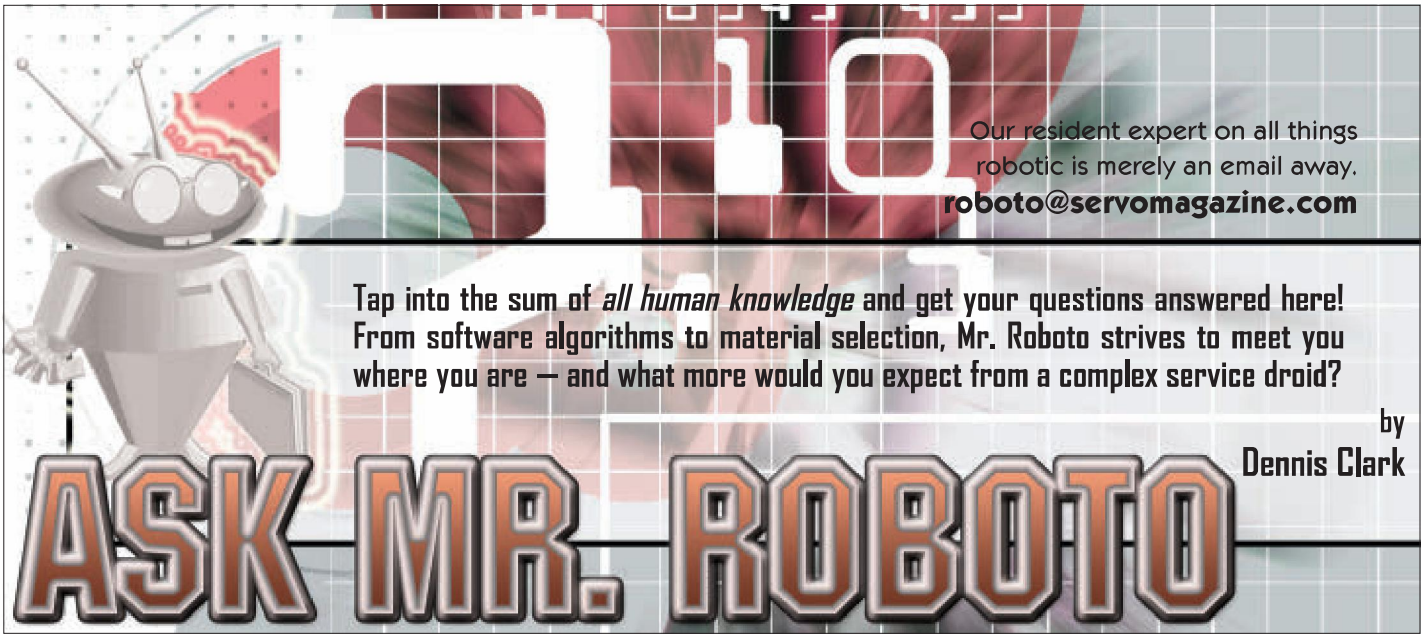
Androids get closer to mastering human expressions.

Creepy Times Two

The closer humanoid bots get to imitating human facial expressions, the creepier it gets. The first example was mentioned in Bots in Brief last month — Diego-san from the University of California, San Diego (**www.ucsd.edu**) with assistance from Japan's Kokoro Co. which built the body; and Hanson Robotics which fabricated the head. The toddlerbot will be used to study sensory-motor and social development in babies. Diego-san mimics the expressions of a one year old child using 27 moving parts in the head alone. Just search "diego-san robot" on YouTube, and you can see him going through emotions ranging from happy, to sad, to surprised, to "I think my strained peas are coming back up."

Perhaps even more disturbing is the Geminoid-DK android which is pretty much an exact copy of its "master" — Prof. Henrik Scharfe of Denmark's Aalborg University (**www.en.aau.dk**). As with Diego-san, the torso is a product of Kokoro. The unit — having no intelligence of its own — is teleoperated.

For example, if the operator opens his mouth, so does DK. Speech is transmitted to a speaker mounted behind him. Amazingly realistic movement is provided by pneumatic actuators, but is limited to the head and upper torso which could be why the professor's wife has stated that she still prefers the original to the copy. Of course, a YouTube search on "geminoid-dk" will turn up a video.
SV



Background

This month's column can be called either "No good deed goes unpunished" or "Curiosity killed the cat." Last month, we looked at controlling an iRobot Roomba platform with an Arduino UNO. Questions came in about problems with my Roomba program, and I was asked if I could make the code more robust and useful. I never shy away from a challenge, so here you go! This month, we'll look at ways of programming that will allow multiple tasks to run (apparently) at the same time.

On Code Reliability:

Between last month and this month, I have discovered many interesting things. Most obvious is that there is more than one kind of Roomba controller! Some are faster or more tolerant of the way you send it SCI commands; others, not so much.

My old Roomba 410 is either faster or more tolerant of lots of SCI commands coming in. My much newer Roomba 440 is not so tolerant. Code that ran great on the Roomba "Red" ran very badly with resets, ignored commands, and "went into the weeds." on my Roomba 440. I found that if I delayed for about 5 ms after each command, that everything was happy! Okay. So be it. That delay isn't too bad if everything runs well when we use it.

Remember from last month I said that my Roomba Red would not run straight? I took the wheel assemblies apart and cleaned the wheel encoders, and that straightened everything right up (pun intended)!

On More Useful Code:

Last month's code was little more than a giant "if/then" statement that did a little bit of robot-ish stuff. This month, I'm going to talk about finite state machines in embedded processors like we use for many of our robots. I'm not going to go into CS-101 programming course language about state machines, so suffice it to say that a state machine allows a program to move from one known condition — or *state* — to another one in a predictable manner.

"Huh!" you say. "So what?" Indeed, that sounds like any program you write, doesn't it? The beauty of a state machine, however, is that your code will *hold that state* while you tell your microcontroller to go off and do something else. (I'll bet that got your attention.)

A long time ago (you know, like back in the 1990s), the robotics pioneer Rodney Brooks wrote about something called robot *subsumption architectures* to create what looked like intelligent behavior in robots. Each element in a subsumption architecture is a behavior that may or may not use sensors for input, and which does use motors (or similar actuators) for output. Higher-level behaviors take control of (subsume) output actuators, even if lower priority behaviors are active. Each behavior is implemented in what Brooks called *modified finite state machines*.

A *modified* state machine in Brooksian terms is a finite state machine (FSM) that uses timers as an input, as well as sensors and states. In short, using a subsumption architecture can make your robot look like it is "alive" and intelligently responding to its surroundings.

In my opinion, what is really cool about this type of behavioral programming is that your robot will start doing things that you did not tell it to do because of the states left and then returned to in your subsumption architecture behaviors. This is called *emergent behavior* and it is what makes our machines look alive.

To learn more about how simple behaviors can look like actual intelligence, I really recommend that you read Valentino Braitenberg's *Vehicles, Experiments in Synthetic Psychology*. This book is a very easy read and will keep you busy for hours

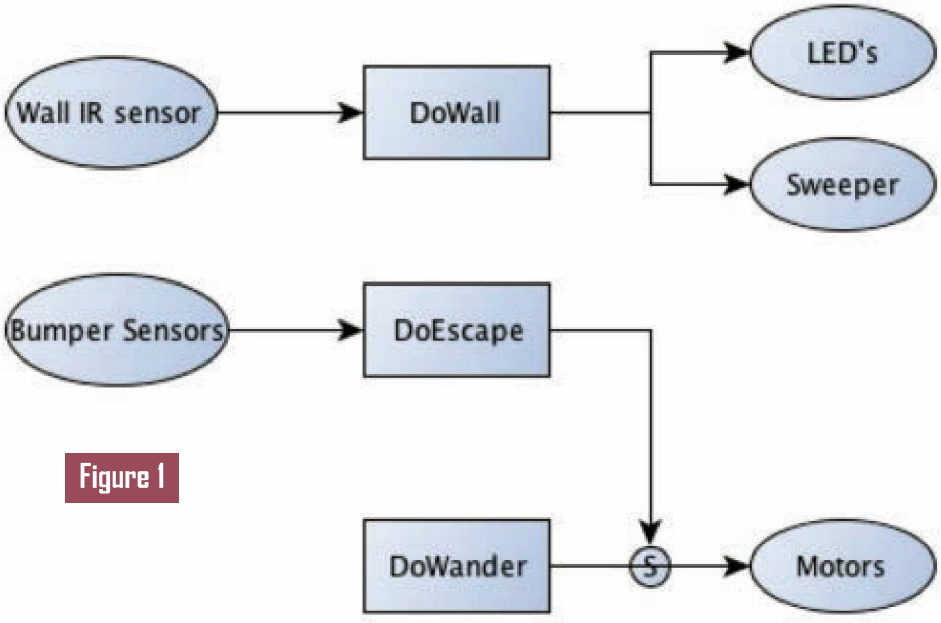


Figure 1

implementing behaviors in your robots. (This way, the next time you hear someone say "Braitenberg Type 2" robot, you'll know exactly what they are talking about!)

I am going to discuss basic subsumption architecture in this article. There are variants on this theme. A behavioral module may have inputs that are suppressed and replaced with other inputs. Or, it may have outputs which can be inhibited to allow outputs from other modules to be used. We will keep this discussion focused on suppressors and timers.

If you are interested in knowing more, there are some books to check out in the **sidebar**. Okay, I've introduced you to new terms and concepts perhaps. Now, you need to know how to design and implement them, so read on.

Designing a Subsumption Architecture

This is going to be a very simple introduction to behavioral programming, involving only a few behaviors. The beauty of the system, however, is that you can layer on more behaviors as you get them to work without modifying existing ones. This means that your behaviors are modular and can be modified without worrying about other behaviors that are currently working fine. Your emergent behaviors might change, however ...

Our new Roomba code will have three defined behaviors to start with:

1. Wander
2. Escape
3. "I see a wall"

The first two behaviors control the drive motors that move the Roomba around. The third one just lights up LEDs and turns on the sweeper motor to tell us that it saw a wall with the wall sensor.

Figure 1 shows what the subsumption network looks like using these three behaviors.

In **Figure 1**, each of the square boxes are behaviors. The *DoWall* behavior stands by itself and has a sensor input and output actuators that are LEDs and the sweeper brush. That behavior isn't all that interesting, but this is how it is represented.

The next two behaviors have subsumption characteristics. The *DoWander* behavior has no sensor input, but it does run the motors. As you can see in **Listing 1**, DoWander just makes the robot go forward — not interesting, but it does something.

The *DoEscape* behavior has a higher priority than *DoWander*, and

Listing 1: DoWanders behavior code.

```
void DoWander(void)
// Define a wander algorithm, or just go straight.
{
  if (priority < wander)
  {
    Serial.println("Wander");
    priority = wander;
    GoForward();
  }
}
```

Listing 2: RoombaBehave1 main Arduino loop.

```
/*
*  The nice thing about state machines is that they allow your
*  main loop to be really simple.
*/
void loop(void)
{
  UpdateSensors();
  DoWall();
  DoWander();
  DoAvoid();
  DoEscape();
  DoPanic();
}
```

uses the bumper sensors as inputs. The output of *DoEscape*, however, goes to the little circle labeled "S." The S stands for suppresses. Therefore, *DoEscape* suppresses the output of *DoWander* and takes over the motors. When *DoEscape* is done with its task, it will return control to *DoWander* or whatever other behavior has the highest currently active priority.

In general, higher priority behaviors are at the top of the **diagram**. That doesn't have to be true, however. All you need to do is look for the behavior that points to the suppressor that has the final say to see which one has the highest priority. When you have all of your behaviors coded, all you need to do in your code's main loop is call everything, and each behavior will handle itself. **Listing 2** shows what our Roomba behavioral program's main Arduino loop looks like. Simple, isn't it?

If you look at **Listing 2** again, you'll note that I have more behaviors there than what this article is talking about. The code (which you can find at the article link as RoombaBehave1.zip) has other behaviors *stubbed* in that can be added later. I am leaving these as an exercise for the reader. (Maybe I'll add some details in a later column, we'll see ...)

I have placed these behaviors in ascending order of priority with the lowest priority called first. Each of these behaviors are *modified finite state machines* (MFSM). There is a standard way to represent FSMs in the programming world.

The *DoWall* behavior is a simple call and has no real state machine; it is always active on each pass through the main loop. Likewise, the *UpdateSensors()* is just a function call that happens each pass through the main loop so that all the behaviors have access to the most recent sensor data. Our first real state machine is *DoWander*.

Figure 2 shows the state machine for that behavior. It is very simple. The first state is the "idle" state where it looks to see if it can even run. The second state goes active on the next pass of the function if the behavior is of a high enough priority to run. As you can see from **Listing 1**, however, this is little more than an if/then statement. Our first real FSM is the *DoEscape* behavior, which tries to get our robot out of a predicament when it runs into something.

Figure 3 shows the FSM diagram for *DoEscape*. Every state that uses the word "Done" is using a timer to decide when the state is finished. This timer is NOT the *delay()* function; that would block the function call until the delay is finished. Instead — as you can see in **Listing 3** — we use the Arduino background milliseconds counter to determine if we are done. If not done, the function exits, leaving the state machine in the same state until the next call of the behavior function.

In the C language, a *switch/case* construct is the simplest way to implement an FSM. Note the timer function *millis()*; I use this as my background timer to know when certain states are ready to change.

Subsumption, behaviors, finite state machines, oh my! This all sounds so technical, but as you can see, the

SELECTED READING

"Cambrian Intelligence, The Early History of the New AI" by Rodney A. Brooks, MIT Press, 1999.

"Mobile Robots, Inspiration to Implementation" by Jones and Flynn, A K Peters LTD, 1993.

"Vehicles, Experiments in Synthetic Psychology" by Valentino Braitenberg, MIT Press 1986.

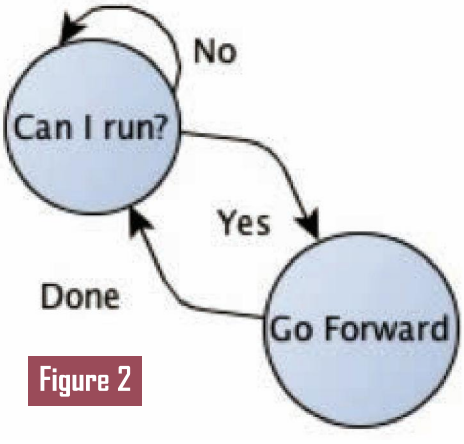


Figure 2

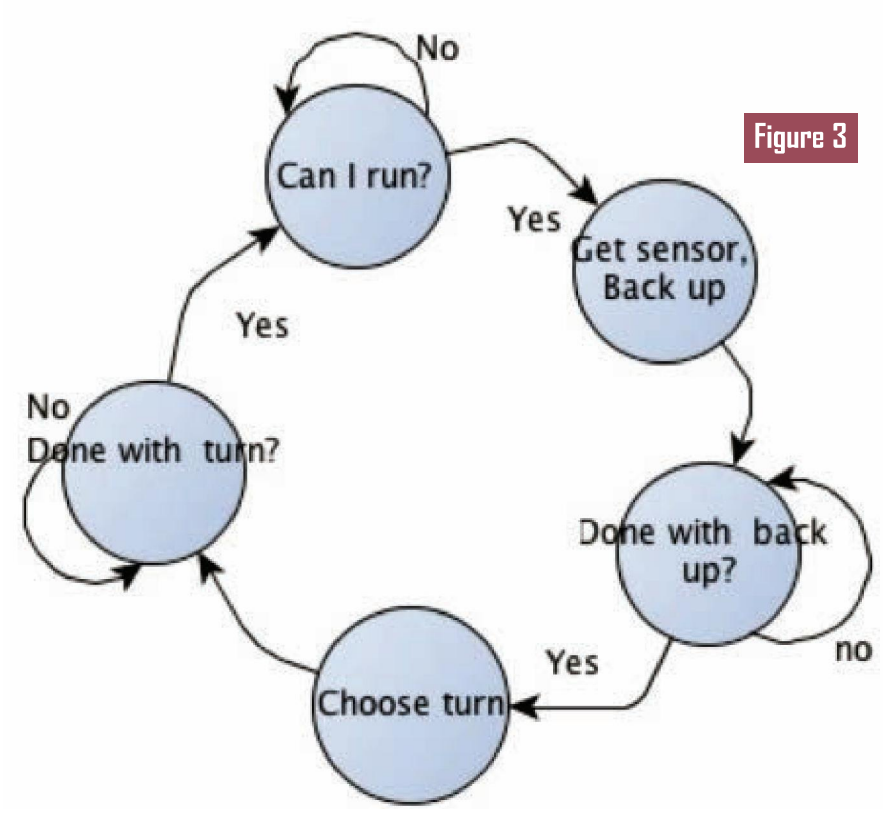


Figure 3

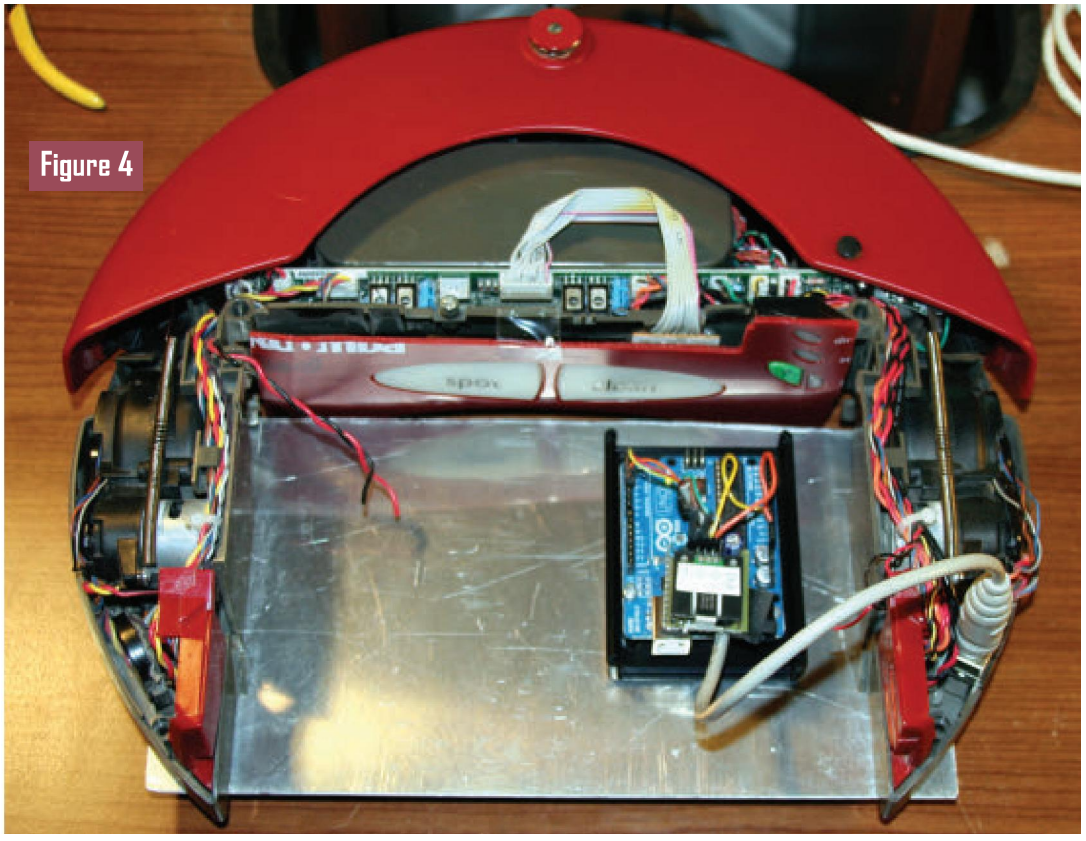


Figure 4

code isn't that complex. This is not to say that it is easy to write! You will spend a great deal of time tweaking and tuning your timing, and choosing your state steps before your behaviors work properly.

The FSM is a basic cornerstone of embedded programming, so if you are inclined to get a job in automation or embedded engineering, put some time in on these concepts — you'll need them!

Back to the Hardware

While working with my Roomba, I got tired of my taped-on dead weights falling off and my Roomba stopping every time it changed directions. So, I

Listing 3: DoEscape behavior.

```
void DoEscape(void)
// Define a bumper response escape behavior
{
  static uint32_t ticker = 0;
  static uint8_t state = 0;
  static uint8_t dir = 0;

  if ((mBumpers == 0) && (state ==0))
  // Don't do anything if nothing to do
  {
    return;
  }
  if (priority <= escape)
  {
    priority = escape;
    switch(state)
    {
      case 0: // record bumper, backup
        dir = mBumpers;
        ticker = millis() + 1000;  // 1 second
        GoBackward();
        GoSong();
        state = 1;
        Serial.print("Bumper dir: ");
        Serial.println(dir,HEX);
        break;

      case 1:             // wait for backup
                          // done
        if (ticker < millis())
        {
          state = 2;
          Serial.println("Do turn");
        }
        break;

      case 2:             // choose turn
                          // direction
        if (dir == RIGHT)
        {
          Serial.println("spin left");
          SpinLeft();
        }
        else if (dir == LEFT)
        {
          Serial.println("spin right");
          SpinRight();
        }
        else
        {
          Serial.println("ahead");
          SpinRight();
        }
         ticker = millis() + 1000;
         state = 3;
        break;

      case 3:             // wait for turn done
        if (ticker < millis())
        {
          state = 0;
          dir = 0;
          priority = idle;
          Serial.println("All done");
        }
        break;
    }
  }
}
```

added a "cargo bay" made of a simple aluminum plate. **Figure 4** shows my new robot with a handy rear deck bolted on.

If you are like me and really don't know how to use a CAD drawing package (yet) to make these simple mechanical plates, **Figure 5** shows my handy-dandy template made from cereal box paperboard and a sharp leather awl that I used to poke screw-hole locations in the paperboard.

It works great and allows me to fiddle with details by just bending the paperboard and using scissors to modify the plates. It is easy to achieve good results with these simple tools.

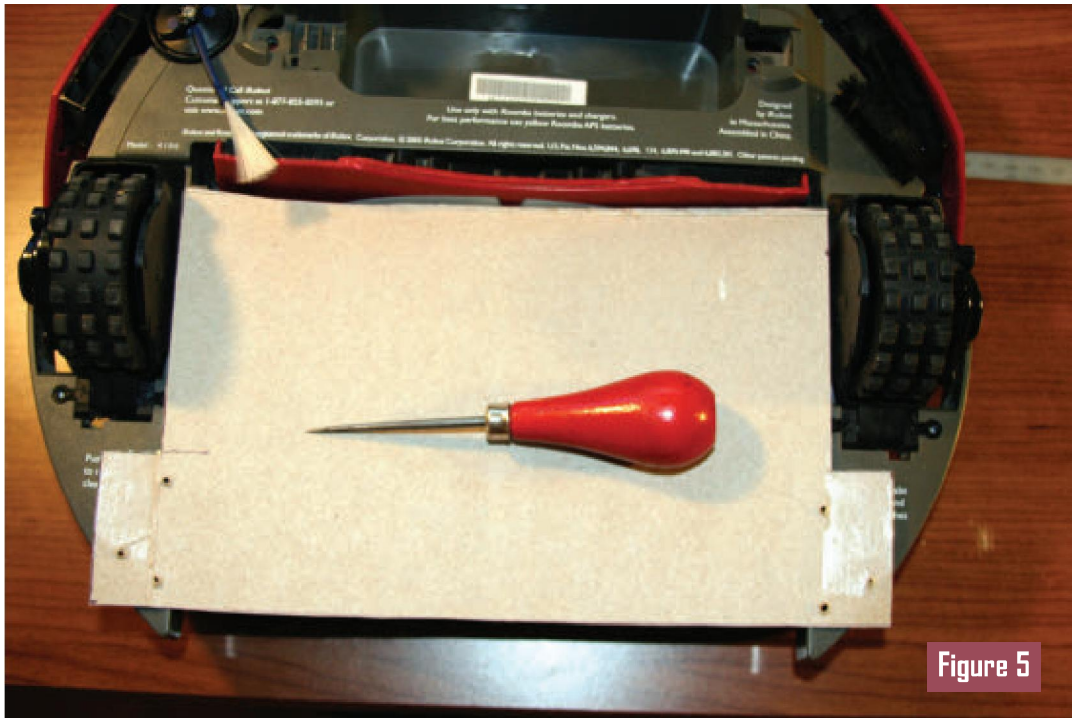


Figure 5

Well, that's all for another month. I love hearing from you, so remember, if you have a robot question, please drop me an email at roboto@servomagazine.com and I'll do my best to answer them. Until next month, keep on building robots! SV

EVENTS Calendar

ROBOTS.NET

Send updates, new listings, corrections, complaints, and suggestions to: steve@ncc.com or FAX 972-404-0269

Know of any robot competitions I've missed? Is your local school or robot group planning a contest? Send an email to steve@ncc.com and tell me about it. Be sure to include the date and location of your contest. If you have a website with contest info, send along the URL as well, so we can tell everyone else about it.

For last-minute updates and changes, you can always find the most recent version of the Robot Competition FAQ at Robots.net: **http://robots.net/rcfaq.html**.

— R. Steven Rainwater

APRIL

4-10 **Alcabot-Hispabot**
University of Alcala, Madrid, Spain
Sumo, mini Sumo, Robocup Jr, Eurobot, Rastreadores, and Velocistas.
www.depeca.uah.es/alcabot

6-7 **Trinity College Fire Fighting Home Robot Contest**
Trinity College, Hartford, CT
Autonomous robots must find and extinguish a fire.
www.trincoll.edu/events/robots

9-11 **DTU RoboCup**
Copenhagen, Denmark
Autonomous robots compete on a course that combines line following and wall following.
www.robocup.dtu.dk

11-13 **Istanbul Technical University Robotic Olympics**
Istanbul, Turkey
Events include line following, mini Sumo, fire fighting, maze solving, and a robot vacuum cleaner contest.
www.ituro.org

11-13 **National Robotics Challenge**
Marion, OH
A wide range of events for student-built robots including Sumo, mini Sumo, pick-and-place, robot construction, robot maze, rescue robot, robo hockey, Bot Ball, and VEX Sack Attack.
www.nationalroboticschallenge.org

13 **Brown IEEE Robotics Competition**
Brown University, Providence, RI
Autonomous robots must navigate a maze.
http://brown.edu/Departments/Engineering/Organizations/Ieee/competition

13 **CIRC Central Illinois Bot Brawl**
Normal, IL
Events include RC vehicle combat, Sumo, line following, line maze, and Checkers on a Sumo Ring.
http://circ.mtco.com

13 **Penn State Abington Fire Fighting Robot Contest**
Penn State Abington, Abington, PA
Regional event for the Trinity College Fire Fighting Contest.
www.ecsel.psu.edu/users/avanzato/robots/contests/firefighting

17-20 **VEX Robotics World Championship**
Anaheim, CA
This year's contest is called VEX Sack Attack.
www.vexrobotics.com/competition

19 **Carnegie Mellon Mobot Races**
CMU, Pittsburgh, PA
Student-built robots compete in the Mobot Slalom and MoboJoust.
www.cs.cmu.edu/~mobot

19-21 **RoboGames**
San Mateo Event Center, San Mateo, CA
Over 50 events for all kinds of robots including BEAM, Mindstorms, Soccer, Sumo, line following, Robomagellan, Bartending, fire fighting, and dozens of others.
www.robogames.net

20 **Mercury Remote Robot Challenge**
Stillwater, OK (and Puebla, Mexico)
Tele-operated robots must navigate a course 100 miles away from the operators.
http://mercury.okstate.edu

20 **RoboRodentia**
California Polytechnic, San Luis Obispo, CA
Autonomous micromouse-like robots must navigate a maze, pick up balls, and place them in a nest.
https://sites.google.com/site/calpolycomputerengineering

20 **The Tech Museum of Innovation's Annual Tech Challenge**
San Jose, CA
This year's challenge: Asteroids Rock!
http://techchallenge.thetech.org

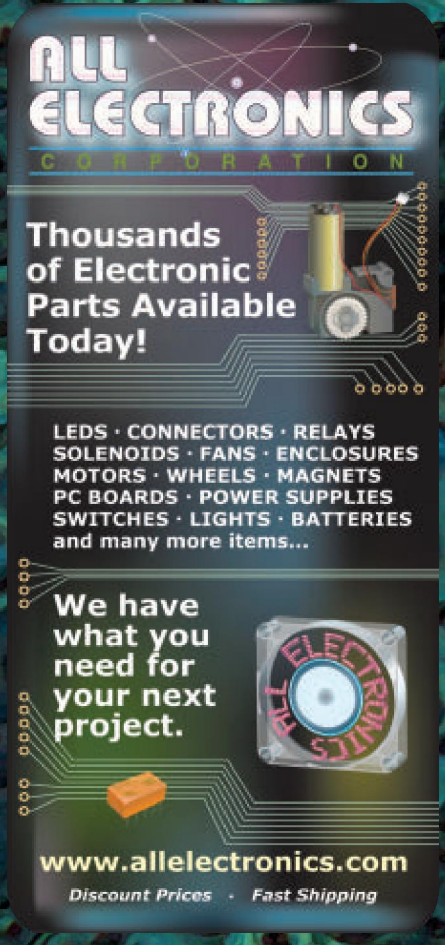
20 **UC Davis Picnic Day MicroMouse Contest**
University of California, Davis Campus, CA
Autonomous Micromouse robots must solve a maze.
www.ece.ucdavis.edu/umouse

24-27 **FIRST Robotics Competition**
St. Louis, MO
Championship for student team-built robots.
www.usfirst.org

25 **Robotdalen Robotics Innovation Challenge**
Eskilstuna, Sweden
Contestants must demonstrate a robot with commercial potential.
www.robotdalen.se/ric2013

26-27 **Greater Philadelphia SeaPerch Challenge**
Drexel University, Philadelphia, PA
Tethered underwater ROV missions.
www.phillyseaperch.org

Robotics Showcase



NEW PRODUCTS

Aluminum Hub Mounts

ServoCity's 0.770 hub pattern has now been incorporated into their new line of aluminum hub mounts. Integrating both 6-32 tapped and .140" through holes (in the hub pattern), and 6-32 tapped base mounting holes provides for countless attachment possibilities. Constructed from 6061-T6 aluminum for superior strength, these mounts won't easily flex or bow. The 1/2" bore makes it simple to incorporate shafting and tubing into a project design. Starting at $4.49, they are offered in many styles and configurations.

Heavy Duty Robot Wheels

ServoCity also recently added a new line of robot wheels to their expanding product selection. The hard surface wheels offer combined high load capacity with low rolling resistance. The foam wheels work well for applications that require high traction and quiet running. The 3" and the 5" wheels are offered in gray, blue, and orange; the heavy-duty and foam wheels are currently offered in black.

All wheels are compatible with ServoCity's line of aluminum hubs and adaptors. Starting at $2.59, these wheels can be used in applications such as designing a mobile robot, a camera slider, or a wheeled platform.

For further information, please contact:

ServoCity Website: **www.servocity.com**

Arduino Compatible chipKIT uC32 Development Platform and Wi-Fi Shield

Microchip Technology, Inc., has announced the jointly developed chipKIT™ uC32™ open source development platform, which is based on Microchip's 32-bit PIC32 microcontroller, as well as the chipKIT Wi-Fi® Shield. Digilent's chipKIT uC32 development board has the same form factor as their chipKIT Uno32™ board, with the addition of larger memory (512 KB Flash and up to 32 KB RAM).

The uC32 board provides a single general-purpose development platform for users to create a wide range of 32-bit MCU-based applications using the free Arduino™ compatible chipKIT IDE called the Multi-Platform IDE, or MPIDE. The chipKIT Wi-Fi shield enables users to implement wireless projects with the chipKIT line of microcontroller

boards, such as the new uC32 or existing Uno32 and Max32™ boards.

The chipKIT platforms enable hobbyists and academics to easily and inexpensively add electronics to projects, even without an engineering background. The chipKIT uC32 board includes a PIC32MX340F512H microcontroller, featuring 105 DMIPS performance, 512 KB Flash, 32 KB SRAM, and 42 I/O pins that support a number of peripheral functions such as UART, SPI, and I²C ports and PWM outputs. Twelve of the PIC32's I/O pins can be used as analog inputs, or as digital inputs and outputs.

The chipKIT Wi-Fi shield makes use of Microchip's agency-certified MRF24WB0MA Wi-Fi module, and provides chipKIT microcontroller boards with the ability to communicate via IEEE 802.11 compatible wireless networks. The Wi-Fi shield also provides a microSD™ card connector for use with microSD Flash memory cards. The chipKIT MPIDE SD library can be used to read/write files stored on the microSD card.

The MPIDE also makes it simple for new developers to introduce their own compatible chipKIT boards. The MPIDE includes libraries — such as Brian Schmalz's SoftPWMServo library — which enables users to generate an analogWrite-style output, as well as an R/C servo output on all pins simultaneously.

The chipKIT uC32 is priced at $34.99 each; the chipKIT Wi-Fi shield is priced at $49.99 each.

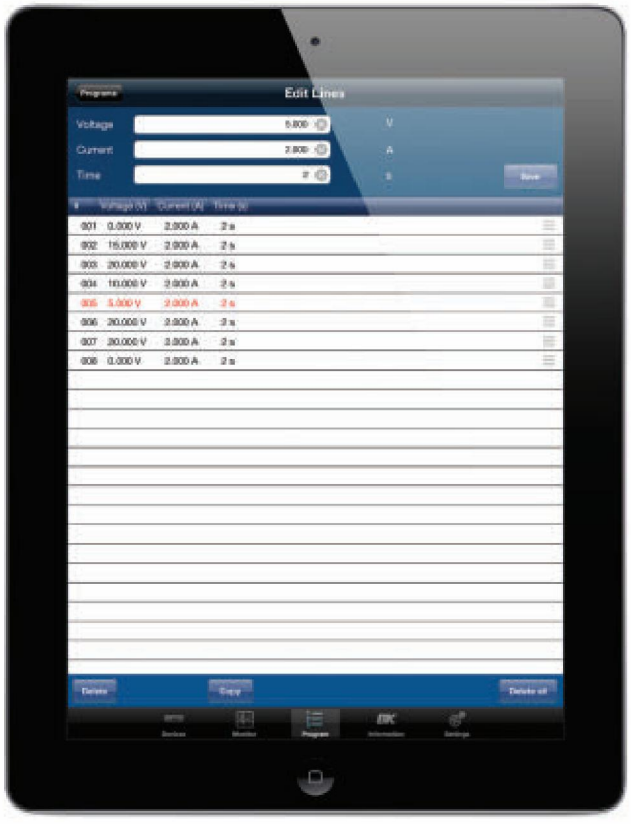
For further information, please contact:

Microchip Technology, Inc.
Website: www.microchip.com

pwrApp for iPhone and iPad

B&K Precision has launched its new pwrApp for the iPad, iPhone, and iPod touch. Adding greater value to its DC power solutions, B&K Precision's pwrApp allows full monitoring and control of network-connected (via WLAN) XLN-GL series of power supplies over local wireless networks.

The primary function of pwrApp is remote operation of all functions of XLN-GL power supplies. Features of the app include live visual monitoring and interactive power supply control, visual data graphing, audible trigger alarms within the



Continued on page 52

EARN MORE MONEY
Get your dream job!

Be an FCC Licensed Radio Technician!

Make up to $100,000 a year and more with NO college degree

Get your "FCC Commercial Radio License" with our proven Home-Study Course!

- No costly school. No classes to attend.
- Learn at home. Low cost!
- No experience needed!
- MONEY BACK GUARANTEE: You get your FCC License or money will be refunded!

Turn your interest in electronics into a high paying career in Communications, Radio, Television, Avionics, Marine, Radar, Safety, Municipal and more... even start your own business!

Call now for FREE info: 800-932-4268 ext. 109

Or, email us: info@LicenseTraining.com

www.LicenseTraining.com

COMMAND PRODUCTIONS, Warren Weagant's FCC License Training
Industrial Center, 480 Gate Five Rd., PO Box 3000, Sausalito, CA 94966-3000

bots IN BRIEF

BLACK HORNET GETS GREEN LIGHT

This helicopter is damn tiny. It's only four inches long (about 10 centimeters) and weighs just 16 grams, but will happily carry a pan-and-tilt camera that streams video back to a base station. It's called the Black Hornet, and it's, well, adorable.

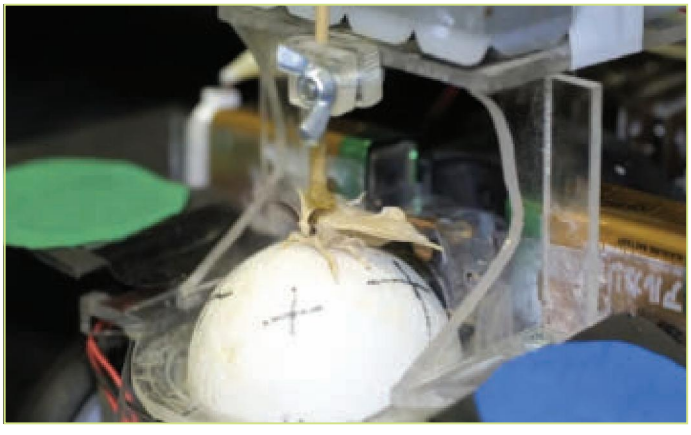
In a military with an emphasis on big expensive robots like predator drones and PackBots, we're starting to see a different trend towards much smaller and slightly less expensive (but still expensive) robots designed for individual soldiers to use whenever they require them. No need to spend time calling in some sort of big fancy aerial remote sensing platform; you can just toss out your own personal recon bot.

The robot is controlled with a handy little thumb joystick thingy, while you watch the video feed it sends back on a tablet. It self-stabilizes to make the flying easier, and there are autopilot modes including GPS waypoint navigation, hover and stare, and pre-programmed search patterns. The Black Hornet is nearly silent, has a range of 1,000 meters, can fly for 25 minutes, and can go from pocket to flying in under 60 seconds.

Black Hornet is definitely adding value, especially considering the lightweight nature of it. British units in Afghanistan use it to look for insurgent firing points and check out exposed areas of the ground before crossing, which is a real asset. It is very easy to operate and offers amazing capability to the guys on the ground.

MOTH BALLS

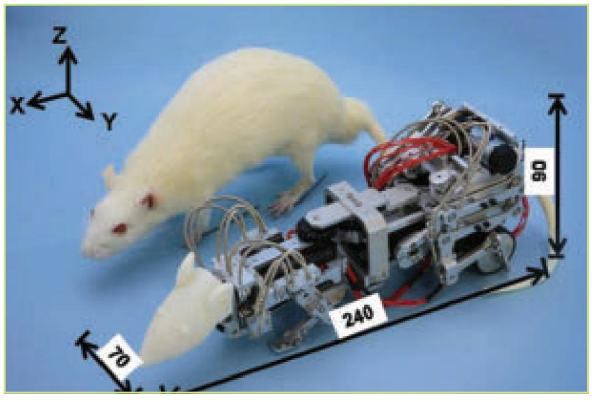
Insect-inspired robots are nothing new, but an insect driving a robot is. Graduate student Garnet Hertz managed to get a cockroach to control a mobile robot back in 2006, but scientists over at the University of Tokyo have changed it up a bit by having a silk moth drive a small two-wheeled bot in pursuit of a female sex pheromone. As with the cockroach, the male moth steered the bot by walking around on a rotating ball — no training required. This isn't just for fun and games of course. The eventual goal of the study is to apply the moth's tracking behavior to autonomous robots so they can be utilized in situations like hunting down environmental spills and leaks.



SAD RATS

You may not be able to tell, but the rat in the photo that isn't a robot is seriously depressed.

Hopefully, you're asking yourself why anyone would care if a rat is depressed, or design a robot that exists only to instill depression in rats. Fortunately, there's a simple answer: We need depressed rats to be able to test drugs that treat depression. Before trying drugs out on humans, they get tested on animals, and that requires a bunch of depressed rats.



There are already some accepted ways to turn a normal rat into a depressed rat. For example, you can force it to swim for long periods, or you can put it into a box and give it electric shocks. Do these things often enough, and you'll be left with one seriously unhappy rat. The problem with these methods is that humans don't usually experience depression that is caused by prolonged swimming and shocks, so researchers at Waseda University in Tokyo are trying to come up with a more accurate model by causing depression in rats with other (robotic) rats.

This rat robot — called WR-3 — is basically an attack robot. It has three modes or "behavior generation algorithms" which include chasing, continuous attacking, and interactive attacking. Chasing means that WR-3 will attempt to maintain a minimum distance from the rat but not actually attack it. Continuous attacking is, well, continuous attacking with aggressive body motions and physical contact. Interactive attacking only attacks the real rat whenever it moves at a set distance.

Experiments on different groups of rats of different ages showed that the most effective means of instilling depression (measured by overall listlessness) was to constantly harass young rats and then intermittently harass them again when they got older. Doing it this way apparently is a better overall model for depression than other methods. So, that's good news for drug testing and bad news for lab rats.

A Novel Method to Develop an Animal Model of Depression Using a Small Mobile Robot is available online at **www.tandfonline.com/doi/pdf/10.1080/01691864.2013.752319** if you want to read it.

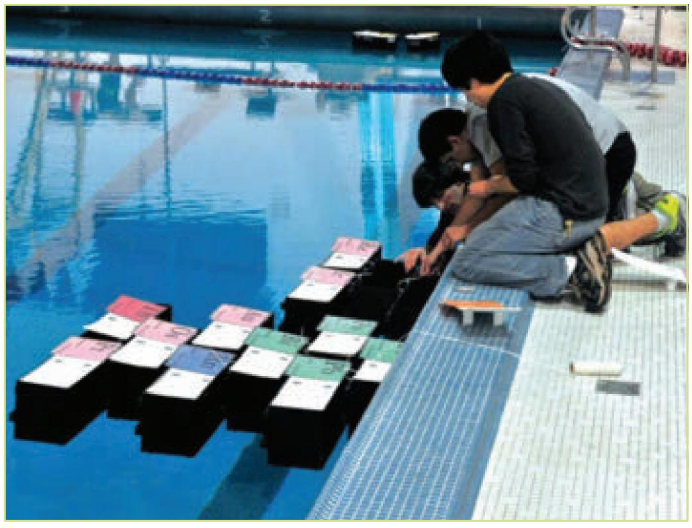
SENSELESS JUGGLING

The Blind Juggler is a robot that is able to juggle a ball on a paddle. The robot does this without any sensing; there are no cameras, no microphones, or any other sensors that could tell the robot where the ball is. That's right. This robot has no hands, no eyes, no sensors, and manages to juggle a ball while swinging back and forth using a single linear actuator. Let me reiterate: no sensors. Imagine yourself juggling a ball blindfolded. It's no easy task!

How does the robot do this? The key elements are the shape of the paddle and how the robot strikes the ball. The developers used mathematical analysis to learn which shape and motion would make "sensorless" juggling possible. What they found was that a slightly concave paddle keeps the ball centered, and that a decelerating motion keeps the ball in the air.

Find out more at **www.blindjuggler.org/the-blind-juggler**.

RO, RO, ROBO BOATS



Yep, robot swarms are now taking over swimming pools.

Mark Yim and his students at the University of Pennsylvania are trying to get this big fleet of small robotic boats to cooperate to form all sorts of useful structures. By coupling together, a large enough swarm would be capable of forming bridges, runways, or even islands. UPenn has over 100 of them — each one of which is controlled with a Gumstix, and uses four separate motors to enable omnidirectional movement and zero-radius turns. According to a recent *Daily Pennsylvanian* article:

"The objective was to get the boats to form a bridge across a corner of the pool and drive a car across the bridge. The week before, the boats successfully configured themselves into an island for one of Engineering professor Vijay Kumar's quadrotor drones to land on."

Now, that's cool!

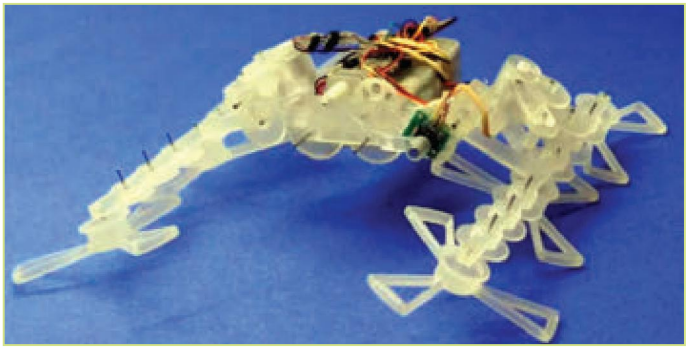
What DARPA would like to see happen is to (eventually) scale this system up to boats the size of shipping containers, since shipping containers are cheap, everywhere, and easy to transport and manage. You could fill a container ship with these modules and dump them all into the water near a disaster zone, tell them you need a runway, and they'd all zip around and form a nice big flat stabilized platform for you. When you're done — or need something else — the modules would unhook themselves and either reconfigure into something different or queue up to get collected and moved on to the next place they're needed.

Oh, and by the way, they named all the boats after period table elements.

BIRD'S EYE VIEW

Scientists and filmmakers love to deploy robots in disguise to spy on real animals. They've sent drones to film herds of elephants and zebras in Africa, armored mobile robots to photograph Kenyan lions, and have used animatronic apes to get close-up footage of bonobos. Recently, the BBC sent 50 special cameras — including robot penguins that can walk and swim — to get inside penguin colonies and capture never-seen footage of the adorable flat-footed, tuxedo-clad birds in their natural habitats.

For their documentary "Penguins: Spy in the Huddle," the BBC hired John Downer Productions to build a variety of special penguin-cams to film three different species. These spycams were not just the typical remote controlled cameras hidden in a fancy case. One of them — the RockhopperCam — is a walking penguin bot with 20 degrees of freedom that is equipped with gyroscopes, accelerometers, and high-def cameras. According to the producers, it can walk over different terrain, stand up if it falls over, and its computer is preprogrammed with 75 different "penguin motions." Other spycams included the EmperorCam, ChickCam, SnowCam, EggCam, Underwater PenguinCam, and the SnowballCam.



STAR POWER

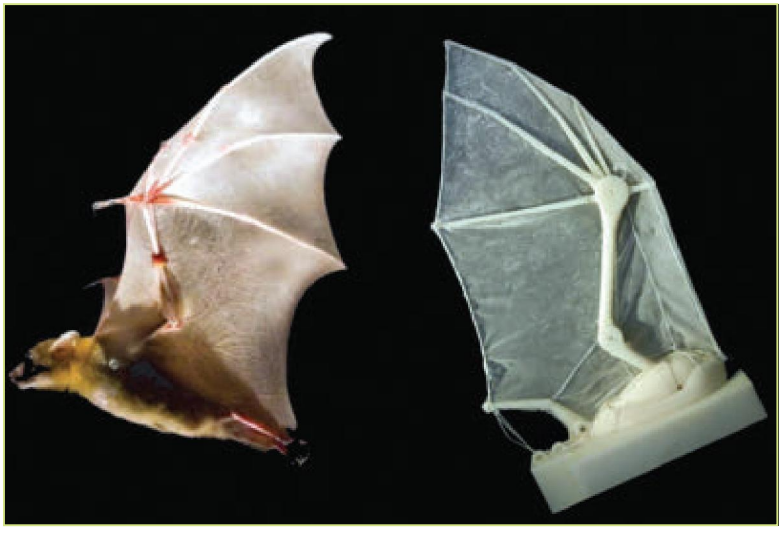
This little guy from UC Berkeley's Biomimetic Millisystems Lab is called STAR for Sprawl-Tuned Autonomous Robot — a six-legged skittery thing just 12 cm in size that can adapt its limbs and its gait to zip over and under obstacles. More info on STAR is scheduled to be presented at ICRA 2013 in Germany in May.

The robot can achieve legged performance over rough surfaces and obstacles using a high sprawl angle, and nearly wheel-like performance over smooth surfaces for small sprawl angles. By changing the sprawl angle, it can climb over obstacles or crawl underneath them. STAR can run at 5.2 m/s (43 body lengths/second, Froude number 9.8) over a smooth surface, which makes it the fastest untethered crawling robot.STAR was developed by David Zarrouk, Andrew Pullin, Nick Kohut, and Ronald Fearing. Check it out at **http://robotics.eecs.berkeley.edu/~ronf/Biomimetics.html#NEW**.

GONE BATTY

Bats are awesome. They are spectacular fliers, and roboticists can learn a lot from them. Problem is, real bats are more interested in catching and eating a bellyfull of bugs than sitting still while biologists fiddle with high speed cameras and x-ray machines and whatnot. The solution, of course, is a robot.

Brown University's robotic bat wing can't quite fly (not by itself, anyway), but it does a great job of pretending. With three servos, seven movable joints, and bones 3D printed to match the real thing, the wing can match the basic flight parameters of bats well enough to allow biologists to measure energy consumption. By tweaking these parameters, it's possible to see what sort of flight techniques make a difference, and then use this data to inform future wing designs for flapping wing UAVs.



One experiment looked at the aerodynamic effects of wing folding. Bats (and some birds) fold their wings back during the upstroke. Previous research from Brown had found that folding helped the bats save energy, but how folding affected aerodynamic forces wasn't clear. Testing with the robot wing shows that folding is all about lift.

In a flapping animal, positive lift is generated by the downstroke, but some of that lift is undone by the subsequent upstroke which generates negative lift. By running trials with and without wing folding, the robot showed that folding the wing on the upstroke dramatically decreases that negative lift, increasing net lift by 50 percent.

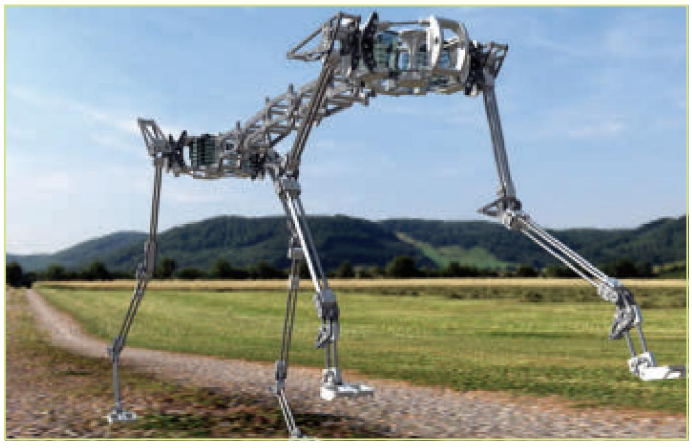
As it turns out, the robot wing breaks a lot, but even that's a learning experience, according to Brown grad student Joseph Bahlman.

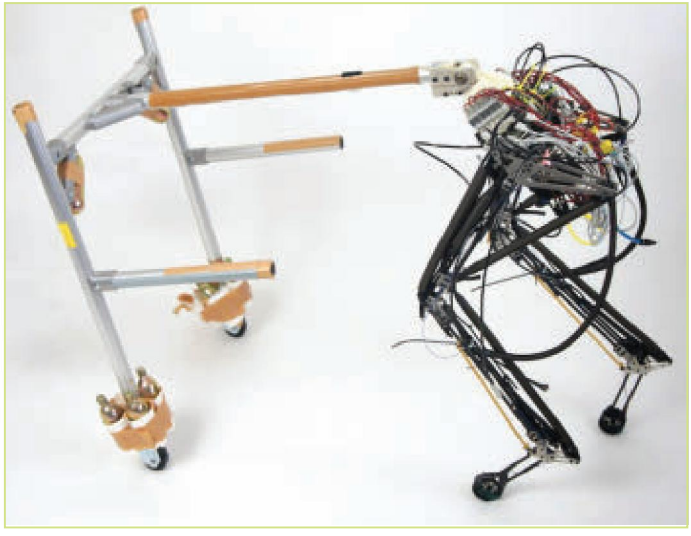
During testing, for example, the tongue and groove joint used for the robot's elbow broke repeatedly. The forces on the wing would spread the groove open, and eventually break it open. Bahlman wrapped steel cable around the joint to keep it intact — similar to the way ligaments hold joints together in real animals.

The fact that the elbow was a characteristic weak point in the robot might help to explain the musculature of elbows in real bats. Bats have a large set of muscles at the elbow that are not positioned to flex the joint. In humans, these muscles are used in the motion that helps us turn our palms up or down. Bats can't make that motion, however, so the fact that these muscles are so large was something of a mystery. Bahlman's experience with the robot suggests these muscles may be adapted to resist bending in a direction that would break the joint open.

The researchers plan to start figuring out all of the complex relationships between flapping frequency, flapping amplitude, flapping angle, wing downstroke time, and wing folding, and also what's energy efficient and what's not. Then, they can start messing with things, swapping out materials and structures to improve on the real thing. If they could teach the resulting robot to survive on insects, they'd be totally set.

PHEUPARD IS THE CAT'S MEOW





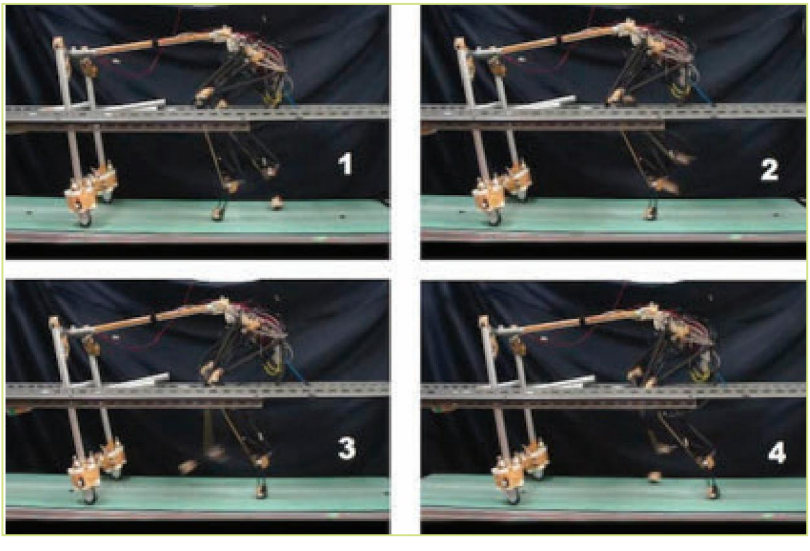
Roboticists around the world are in the process of reverse-engineering the anatomical construction of cheetahs and other cats in an attempt to develop faster and more agile legged robots. The latest project — dubbed the Pneupard — hails from Osaka University. Although still early in development, the new biomimetic platform stands out from some of the others through its use of pneumatic artificial muscles as its primary means of locomotion.

The Pneupard is a creation of researchers Andre Rosendo and Shogo Nakatsu, under the supervision of Kenichi Narioka and Professor Koh Hosoda at Osaka University's Graduate School of Information Science and Technology. The team has worked extensively with pneumatic artificial muscles in the past.

Pneumatic artificial muscles can be made from a rubber tube sheathed in nylon, and contract much like the real thing when filled with air. They can pack a lot of power in short bursts and are also highly flexible and impact-resistant, giving them a lifelike quality that is often missing in robots powered by electric motors. The muscle contractions — when combined with limbs that accurately replicate the length, forces, and range of motion of a real cat — generate a lifelike movement.

Rosendo, the project leader, explains that in the current prototype, each hind limb has eight active muscles and the fore limbs (which are under construction) will have six muscles. "The spine will also have muscles, performing flexion and extension," he says. "The main idea of this project is to create a biomimetic platform where we can replicate feline structure to better understand how they can excel in so many different areas in locomotion. After grasping their secret, it would be possible to apply that knowledge to future robots."

Rosendo notes that the researchers haven't pre-programmed the muscles by hand; the muscles actually activate based on EMG signals recorded from a cat walking on a treadmill. The only sensors used by the robot are force sensors in each foot. "The walking is robust enough to withstand random disturbances, including asymmetry between the right and left caused by the artificial muscle construction method," Rosendo explained. "We are compelled to believe that the musculoskeletal structure found in animals is 'intelligent' enough to compensate for disturbances without requiring a sophisticated control method."



INSIDE(S) LOOK

A $1 million bionic something or other is now on display at the London Science Museum. It's not really a man and it's not really a robot. It's not really a cyborg, either. What it is, actually, is a showcase of all the artificial systems that can be installed in humans to try and fix things that aren't working. It's also a look towards the future of augmented biology — all stuffed into a human form. "Rex," was put together by Shadow Robot Company.

Internal stuff used includes the following:

Arteries and Trachea: Nanocomposite polymers seeded with stem cells. Developed by Royal Free Hospital in London; patients have successfully received implants.

Blood: A paste of plastic molecules suspended in water with similar oxygen binding and releasing properties as the hemoglobin in red blood cells. It was developed by the University of Sheffield and is currently undergoing laboratory development.

Kidneys: An immune stable nanoscale filtration system coupled with renal tubule cells in an implantable cartridge. Developed by UCSF; should be ready for clinical trials in three to five years.

Heart: Battery powered, self-contained, total replacement system. Developed by SynCardia Systems; it has been approved for implantation in humans by the FDA.

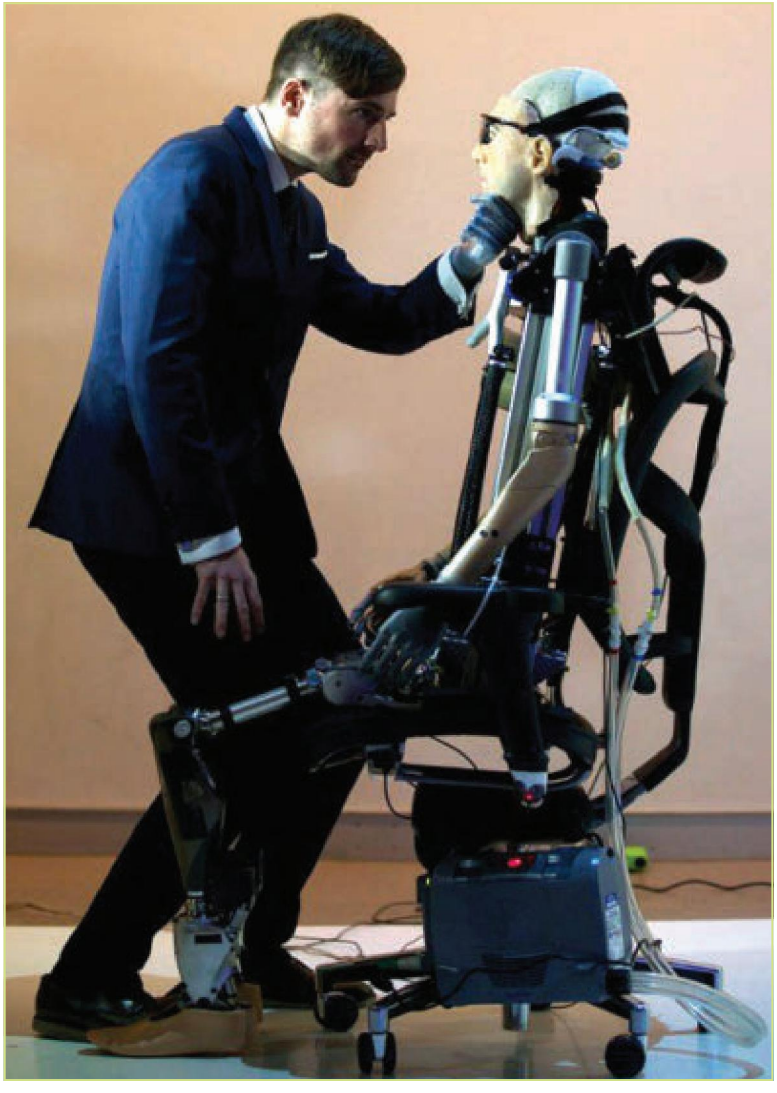
Spleen: Magnetic nanoparticles pull pathogens out of the blood and into a saline solution through a porous membrane. Developed by the Wyss Institute, it is undergoing testing for battlefield triage deployment.

Pancreas: Circuitless glucose-sensitive insulin gel releasing implant. Developed by De Montford University; currently under development.

Lungs: Portable blood/air mass exchanger which can remove CO_2 from and add oxygen to the blood of active patients through gas permeable, surface-coated hollow fiber polymers. Developed by Haemair; a prototype is in laboratory testing.

Eyes: External video camera sends images through a processor which transmits them wirelessly to a receiver mounted on the eye. An array of electrodes placed directly onto the retina sends signals along the optic nerve, which the brain learns to interpret as images. Developed by Second Sight, and currently undergoing international clinical tests.

There are also lots of other pieces of cybernetic hardware, like arms, hands, hips, knees, and feet, and even an external powered exoskeleton.



Cool tidbits herein provided by www.botjunkie.com, www.robotsnob.com, www.plasticpals.com, http://www.robots-dreams.com, and other places.

COMBAT ZONE

Discuss this section in the *SERVO Magazine* forums at http://forum.servomagazine.com

Featured This Month:

26 *BUILD REPORT: Mini Bot Hockey Bots — Part 3*
by Pete Smith

29 *Is This the Future of Combat Robotics?*
by Pete Smith

30 *PARTS IS PARTS: Product Review: FingerTech Robotics Power Switch*
by Mike Jeffries

31 *CARTOON*

32 *EVENT REPORT: GCRS 11 — Robots, Birthday Cake, and Off-Road Trucks!*
by Andrea Suarez

www.servomagazine.com/index.php?/magazine/article/april2013_CombatZone

BUILD REPORT:

Mini Bot Hockey Bots — Part 3

● by Pete Smith

In Part 2 of this series, I discussed how I made the chassis and drive train. In this part, I will show you how I wired the bots and set up the transmitter and receiver.

Oftentimes, wiring in a bot can look like a rat's nest (**Figure 1**), and be rather daunting to someone who has never built a robot before. Fortunately, it's not as complicated as it looks.

The wiring diagram (**Figure 2**) shows where each wire goes to. Power flows from the battery on the left to the power on/off switch first. Since this is a hockey bot, I used a cheap commercial on/off switch that had a built-in LED indicator designed to run on 12V.

A combat bot should use switches such as those offered by Team Whyachi, or simply use a wire link as these are much less likely to get knocked to the "off" position in an impact, or fall apart.

I carefully checked out which terminal on the switch should be connected to the battery; it is the power out and the last one which is the ground for the LED indicator. In this switch, they are marked +, A, and the ground symbol, respectively. You could use spade terminals to connect the leads but I soldered them on as shown in **Figure 3**.

Try to stick with red for the positive (+) wires and black for the negative return (-) wires. This reduces the chances you will short out the battery or connect the ESCs up the wrong way which will often result in them being damaged.

The battery has to have its own connector so that it can be removed from the bot for safe charging. However, some builders like to solder all the other connections directly. This saves weight and space, but makes replacing any failed component that much harder — especially

when under pressure during an event.

I prefer each part to have its own connectors so that it can be quickly unplugged and replaced. In this bot, I used Deans type connectors for the power connections and 2 mm gold for the connection to the drive motors.

I modified the drive ESC (I used Banebots 12-45s, in this case) by removing the heat shrink and unsoldering the single motor leads, then replaced them with two smaller gauge wires on each side. I recovered them with new heat shrink tubing (**Figure 4**). This makes it easy to use one ESC to power both motors on one side of the bot.

Make sure you connect the right-hand ESC to the motors on the right and the left-hand ESC to the motors on the left. It's easy to plug them into the wrong motor since all the leads can get mixed up in the tight space between the ends of the motors.

I'm using a new radio in this bot: the HobbyKing HK6S (**Figure 5**). This is a simple entry-level 2.4 GHz spread spectrum radio with built-in mixing. It looks rather odd since it has no external antenna, but this is an advantage in my application as the kids who will be using the bots often end up breaking them off!

The radio failsafes correctly for combat use, and the channels are easily reversed using the switches on the front (**Figure 6**). If you use this radio in a combat bot, reverse the throttle channel or you will find the throttle is full on in the down position; most ESCs will not power up this way. It uses eight AA alkaline batteries (not included) and is already bound to the supplied receiver. Set up the switches on the radio as shown in **Figure 6**.

The signal leads from the ESC plug into RX Channels 1 and 2 which are Aileron and Elevator, respectively (**Figure 7**). The black wire in the signal leads goes to the outside as shown.

Set the bot up on a block of wood (or something similar) so that its wheels cannot touch the ground. Switch the transmitter on first, then switch on the bot. When you push the left lever on the radio forward, the wheels should all turn in the same direction that would make the bot drive forward if it was on the ground.

If any of the wheels go in the wrong direction, swap around the leads going to that motor (this is where you'll find the 2 mm connectors make life much easier than if you had soldered them on).

Once you have all the wheels turning in the right direction for forward and back, push the lever to the left; the right side wheel should turn forward and the left side wheel

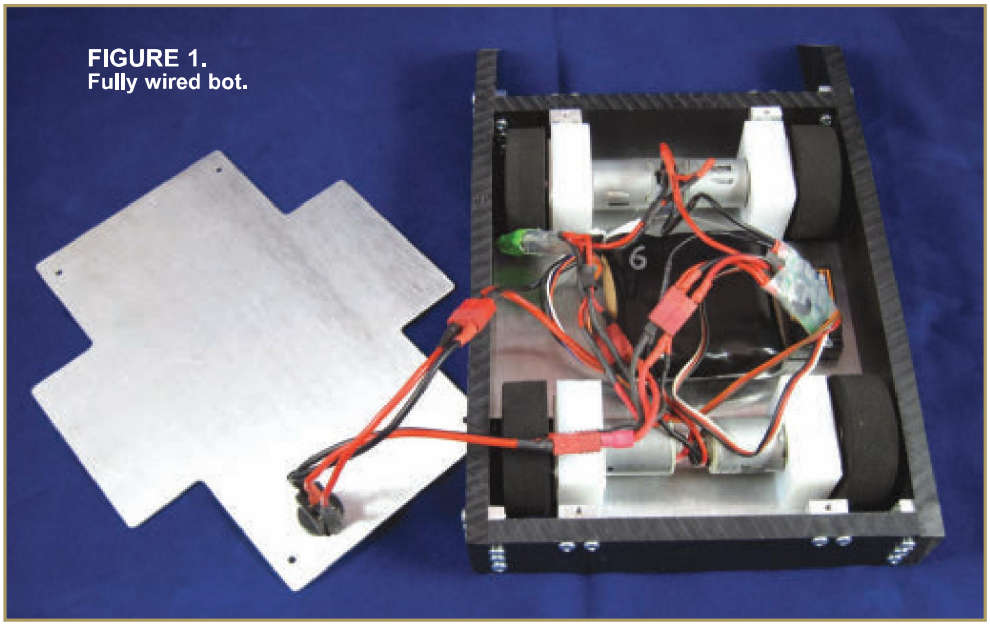


FIGURE 1.
Fully wired bot.

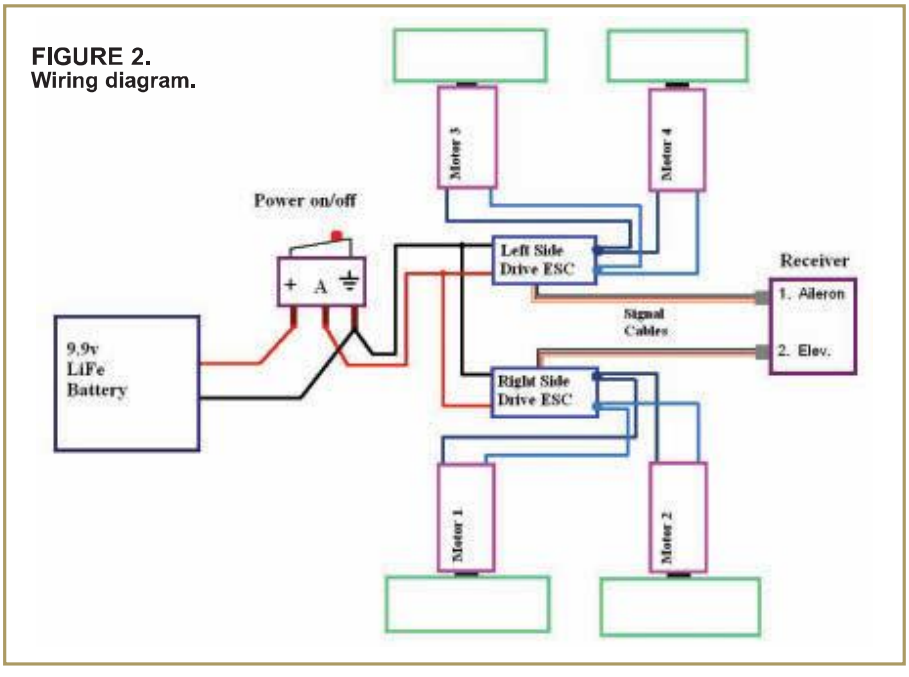


FIGURE 2.
Wiring diagram.

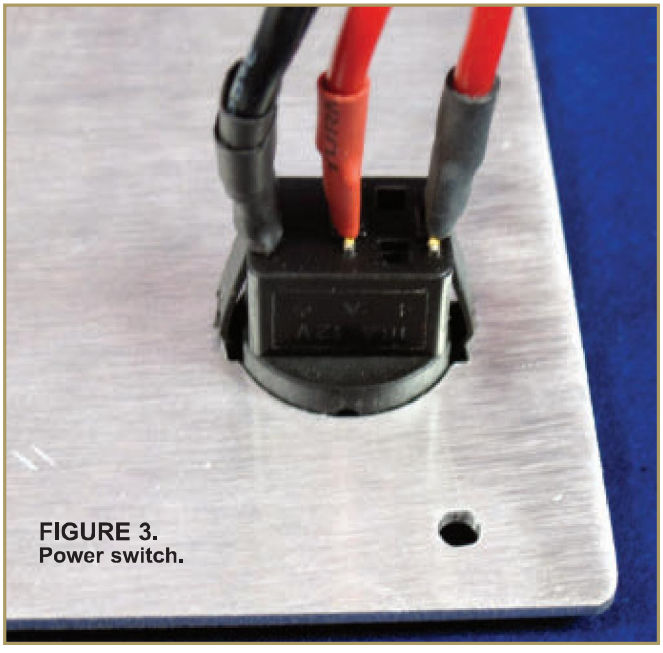


FIGURE 3.
Power switch.

FIGURE 5. HK6S transmitter.

should reverse. This will make the bot turn to the left.

If, however, the bot turns right instead, swap the signal leads in the receiver so that the wire in Channel 1 is now in Channel 2, and vice versa. You may have to reverse the leads on the motors again, but you should be able to get them all running in the correct direction when you move the lever forward and back and side to side.

Tidy up all the wiring so it's clear of the wheels using tiewraps, etc. You are now ready to go drive your bot.

This robot proved to be quite fast and powerful, but still easy to drive. In fact, it's ideal for its designed task: to let kids have fun playing bot hockey.

I've completed two of the bots. Now, I just have to get the rest finished before summer camp starts! **SV**

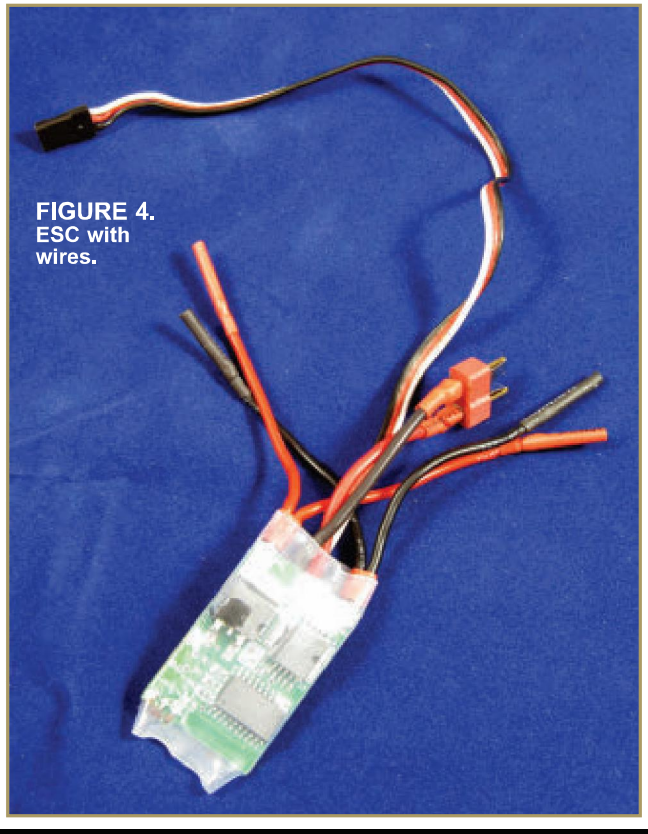


FIGURE 4.
ESC with wires.

FIGURE 6. Switch settings.

FIGURE 7.
Receiver.

Is This the Future of Combat Robotics?

● by Pete Smith

There have been whisperings and rumors of a revolutionary new drive system for over a year now, but details have been hard to come by until our photographer got a picture at a recent late night test session in a barn in rural Pennsylvania. Our contact (who wished to remain anonymous) could give us only a few details of what she called "hoverbots."

The bots use linear induction superconducting perfect diamagnet motors to "hover" and maneuver a few inches above a pyrolitic carbon paneled floor. The bots still have wheels because — at this time — they are still required to give a translation over the panels to initiate the lift motors.

It's quite startling to see the bots start off across the floor and then suddenly leap about a foot into the air before steadying at about three inches above the arena floor.

The bots appear to be able to move about every bit as well as conventional bots, but never actually contact the floor or even the bump rails. They even appear to be able to push with a considerable force.

These new hoverbots will make life very hard for conventional horizontal spinners and low wedges. With the wheels only required for initial lift, you can't even disable them by the usual wheel-ectomy!

There is, however, one thing that might keep them out of the mainstream of robotic combat. That is the recent shortages of the superconducting unobtainium caused by the recent revolt by the indigenous population in Pandora.

If that problem is resolved, the bots will have made their first public appearance at the London Open event on April 1, 2013. SV

Photos by Brian Benson (**www.bensonpv.com**) and Rory Duncan (**www.roryandstaci.com**).

FIGURE 1. Hoverbots in action.

FIGURE 2. Hoverbots — no strings.

PARTS IS PARTS:

Product Review: FingerTech Robotics Power Switch

● by Mike Jeffries

FIGURE 1. An example of a removable link used in a 1 lb fighting robot.

For as long as I can remember, there hasn't been a power switch designed for the 150 g to 3 lb combat robot classes that is readily available. Most Insect class bots I've seen have used variations of a removable link concept, where a connector is used to split the line and the mating connector has the wires connected; when the mating connector is plugged in, the circuit is complete and the robot is powered.

This method works, but in practice is often slow and awkward to use. This method also means that there is the potential for the mating connector to get lost before a match or get knocked loose during a match.

Another option that I have seen is off-the-shelf slides, pushbuttons, and toggle switches. Once again, these work, but they tend to be fragile; without a mechanical lock, they are capable of being switched off purely from sudden acceleration.

I only started building Insect class robots a few years ago, but right from the beginning I knew I wanted a better power switch option. I even went to the effort of designing and building my own. However, they were too expensive to make on a commercial scale, so when I found out about the new switch FingerTech Robotics was working on, I knew I had to take a look at it.

The design is simple, light, and compact. As far as simplicity, there are only five distinct parts: two mirrored housing blocks, one through-hole copper tab, one tapped copper tab, and a screw. As far as weight is concerned, the switch I tested weighed in at 2.2 g.

The switch is also small enough to fit into most Insect class robots at only 0.5 x 0.5 x 0.25" for the main body. With the price coming in under $10, it's also a very good deal.

As delivered, the switch can either be used with 2-56 screws or tapped for 4-40 screws. The nylon housing taps easily and should be quite durable, as long as the screws are not over-tightened. For my tests I chose to tap the holes for 4-40 screws to allow easier installation and save some internal space in the 150 g test bot.

The switch was mounted flat in the chassis with the bolt head accessible from the side of the robot. The switch was mounted in this fashion for two reasons. The first is that it means the switch is easily accessible if the robot is inverted. The second is that the internal chassis dimension is 7/16", which means the switch would not fit if it were mounted vertically.

The switch has performed flawlessly in my own testing and all of the other testing that I am aware of. In addition to robot testing, the switch has also been load tested up to 57A for five minutes without failure. The switch was tested at 76A; however, the 12 gauge wires unsoldered themselves after just under two minutes. The switch itself failed after five minutes at 95A.

Given that the switch is intended for robots weighing up to 3 lbs, this provides a huge margin for even the highest powered 3 lb robots. The non-obvious

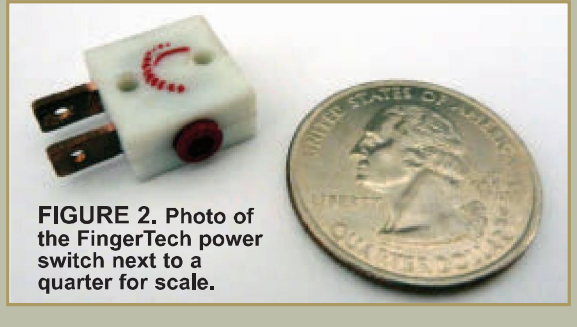


FIGURE 2. Photo of the FingerTech power switch next to a quarter for scale.

feature that greatly contributes to the reliability of the switch is the slightly undersized hole for the screw head. This arrangement results in a head that naturally resists rotation.

When this is combined with the contact method allowing for slight compression, it means you've got two forces acting to prevent the switch from accidentally turning on or off.

After testing the switch, I intend to use it in all of my future Insect class robots. SV

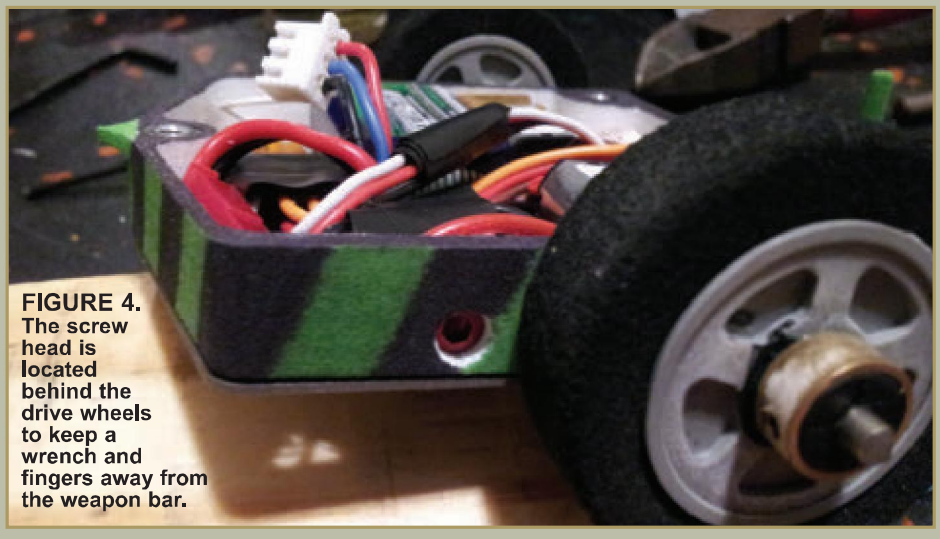


FIGURE 4. The screw head is located behind the drive wheels to keep a wrench and fingers away from the weapon bar.

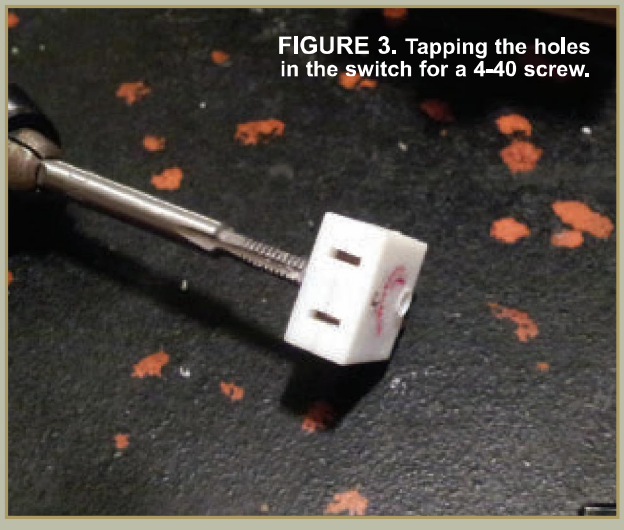


FIGURE 3. Tapping the holes in the switch for a 4-40 screw.

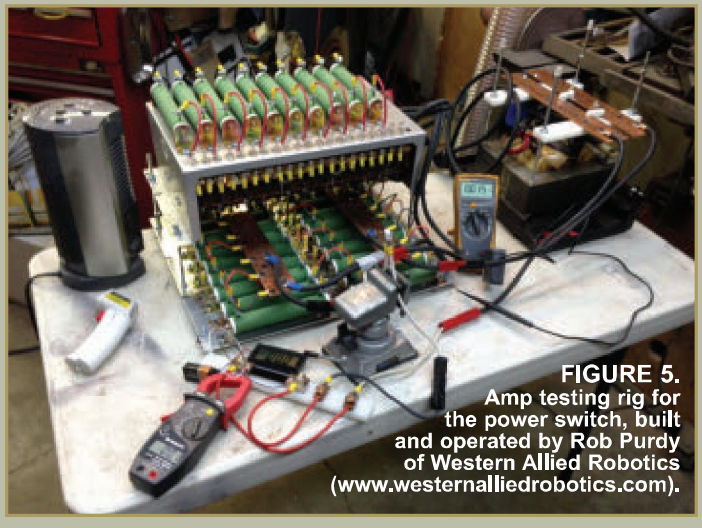
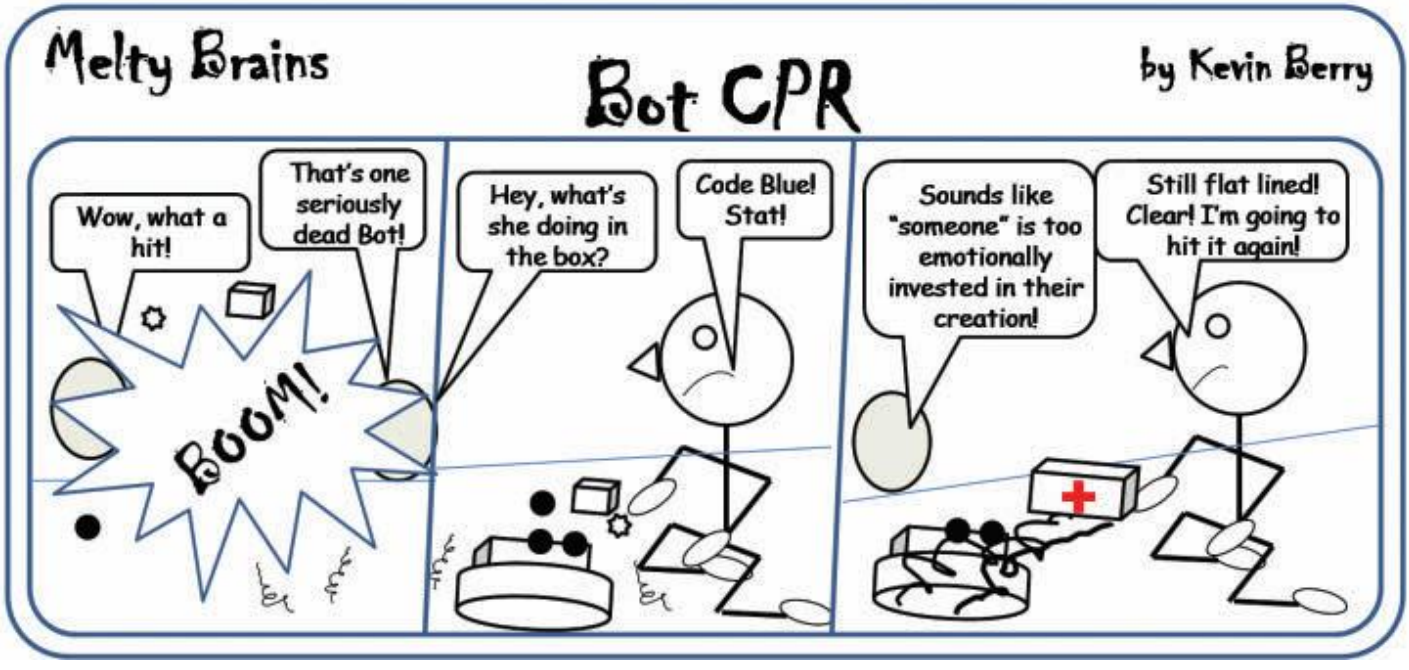


FIGURE 5. Amp testing rig for the power switch, built and operated by Rob Purdy of Western Allied Robotics (www.westernalliedrobotics.com).



EVENT REPORT:

GCRS 11 — Robots, Birthday Cake, and Off-Road Trucks!

● by Andrea Suarez

GCRS 11 continues its series of events hosted by Jim Smentowski, owner of The Robot Marketplace. The events give Florida Insectweight robot builders a chance to get together for a day of robot battles with some new competitors, as well as to continue some long-fought rivalries!

The Antweight class started with Busted Nuts' Hoodoo vs. Ram Robotics' Calamity Kid. Hoodoo tried to use its hinged wedge to push Calamity Kid early in the match, but Calamity Kid's drum soon destroyed one of Hoodoo's wheels, followed by a second wheel, and then finally ripped off a third wheel as Hoodoo flew across the arena.

Ram Robotics continued winning with their second Antweight Capricant, as it beat Duke Robotics' four-wheel drive wedge, Blue Devil, when it pushed it against the wall and got it jammed between the floor and the Lexan. Busted Nuts' WhipperSnapper tried some new design changes to increase the durability of its direct drive weapon set-up, fighting its way through to the semi-finals for a 3rd place win.

Capricant and Calamity Kid battled it out in the finals in a show of driving skill, as both robots came dangerously close to falling out of the drop-out zone. After the full three minutes, a close judge's decision declared Capricant the Antweight winner.

A new competitor to the Antweight division was Matt Spurk's five year old son, Cayden, with his very first bot. Although he didn't win the event, he took every chance he could to grudge-match the other competitors. "With very close father supervision," he has already built a new robot for the next event that he has named Speedy. Watch out, GCRS!

Busted Nuts Robotics' Pissed Off Unicorn faced off against Duke Robotics' Thresh to start the Fleaweight bracket. Both robots spun up to full speed and met in the center of the arena, taking direct weapon hits that launched the tiny robots against the walls. One big hit broke one of Thresh's weapon rails, as Pissed off Unicorn

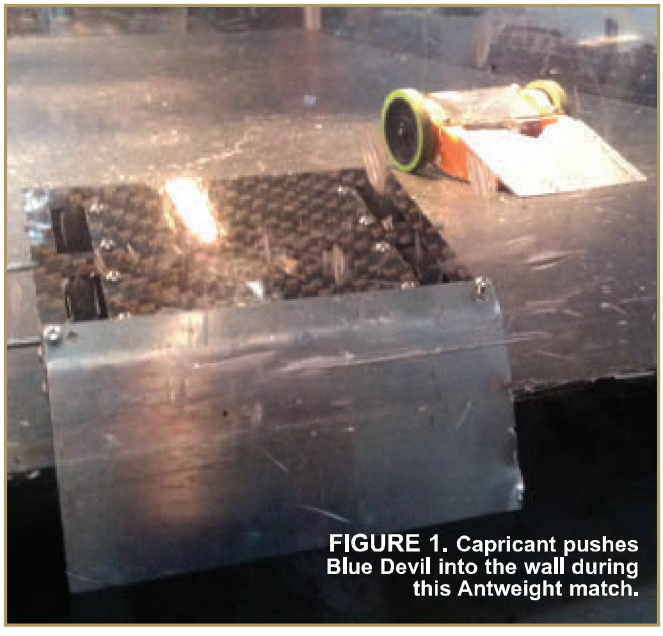


FIGURE 1. Capricant pushes Blue Devil into the wall during this Antweight match.

FIGURE 2. Antweight Capricant almost pushes Calamity Kid out through the drop- out zone!

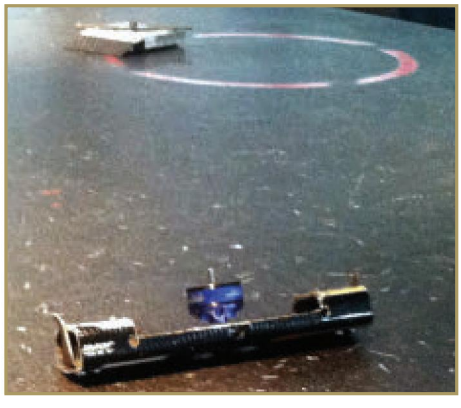


FIGURE 3. Fleaweight BoomStick loses his weapon versus I Kid.

TABLE 1 - GCRS 11 WINNERS

	FLEA	ANT	BEETLE
1st:	I Kid	Capricant	Steve
2nd:	Pissed Off Unicorn	Calamity Kid	DeJa Voodoo
3rd:	inVertigo	WhipperSnapper	Voodoo Magic

FIGURE 4. Pissed Off Unicorn takes its revenge against I Kid during the Fleaweight rumble.

FIGURE 5. Sam McAmis shows the damage to his team's robot, Spare Parts, after the Beetleweight match against DeJa Voodoo.

took advantage and threw Thresh out of the drop-out zone with a single hit from its eight-inch horizontal blade.

Pissed Off Unicorn continued to fight inVertigo, but inVertigo's exposed wheels soon fell victim to the powerful weapon, forcing inVertigo to fight its way through the loser's bracket to the semi-finals, placing 3rd overall. Matt Spurk's Boom Stick surprised the crowd as its horizontal spinning bar took flight and came off his robot! Pissed Off Unicorn's luck ran out in the finals as Ram Robotics' I Kid took the win for the Fleaweight trophy.

The Beetleweight bracket got off to an exciting start as Busted Nuts' DeJa Voodoo flipped Spare Parts high in the air early in the match. New members of the team built Spare Parts based on RamVac's design, and managed to use their robot to get DeJa Voodoo up against the drop-out zone. Just as DeJa Voodoo was being pushed out, its drum finally got a hit on Spare Parts and launched it across the arena.

After almost a full match of hit after hit, both robots were simultaneously immobilized and the judges declared DeJa Voodoo the winner. The next match paired Voodoo Magic — also from Busted Nuts Robotics — against TeamPyramid's Box. These robots have faced off in previous GCRS competitions, and Box tried to keep its wedge pointed at Voodoo Magic's disk. This strategy kept Voodoo Magic's vertical disk ineffective, until a hit to Box's side threw Box high in the air and over the arena barrier.

DJ Spinny fought Walla Walla in the next match, both with horizontal weapons. Although Walla Walla had competed previously in GCRS, Paul Grata redesigned the robot for this event: "I changed the base design to be more of a frame than a solid piece of metal, which resulted in a PVC top-plate. I switched to Pololu's for the drive and changed from a friction based brushed weapon system to a belt based brushless, with the weapon motor on top to maintain a low frame profile."

The finals of the Beetleweight division were the highlight of the event, pairing Ram Robotics' Steve

FIGURE 6. Robot pile-up in the Beetleweight rumble!

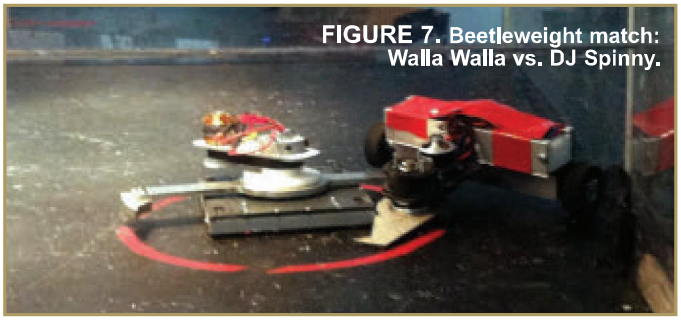


FIGURE 7. Beetleweight match: Walla Walla vs. DJ Spinny.

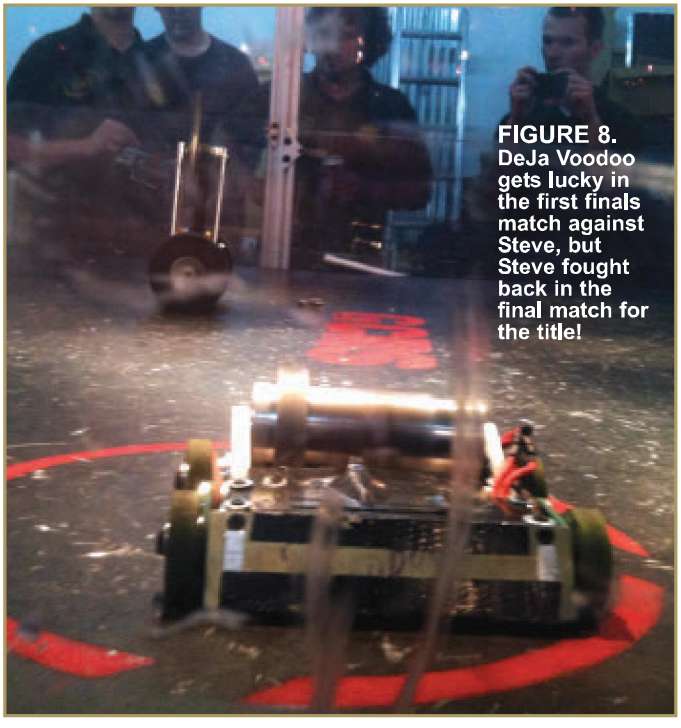


FIGURE 8. DeJa Voodoo gets lucky in the first finals match against Steve, but Steve fought back in the final match for the title!

FIGURE 9. Joel Cooper and Mike Gellatly receive their Beetleweight awards, 1st and 2nd place, respectively.

against Busted Nuts Robotics' Deja Voodoo. Steve was a newcomer to GCRS, but had recently won ComBots.

Steve's driver, Joel Cooper, is finishing his second year of graduate school at USF toward a PhD in Mechanical Engineering with a focus on acoustic forces and cell/tissue engineering. He explained that Steve was originally built for USF's Engineering Expo in 2012 with a designed based off of the heavyweight robot, Last Rites.

FIGURE 10. Elizabeth McAmis made robot-inspired birthday cakes of Walla Walla and Voodoo Magic.

At 11,000 RPM, "it demonstrates the way kinetic energy can be stored and transferred in a way that is exciting to kids and adults alike."

Steve had put DeJa Voodoo in the loser's bracket earlier in the day in a match that resulted in a significant gash high on the arena wall. "GCRS was the only competition where Steve had full armor on his backside. In previous competitions, we used a mixture of pool noodles and gorilla tape to keep wires/speed controllers inside," explains Joel.

This proved critical against DeJa Voodoo, as both robots delivered huge hits during their three matches of the event. In the first finals match, DeJa Voodoo's hit immobilized Steve by forcing the robot onto its side, giving DeJa Voodoo another chance at the victory.

After many violent hits in the final match, Steve delivered a direct hit to DeJa Voodoo's battery for the win!

The robot fighting concluded with a rumble for each weight class. Antweight Hoodoo decided to join in the fun of the Beetleweight rumble and outlasted most of the bigger robots — despite considerable damage!

The event winners took home some nice trophies courtesy of Ram Robotics, and Robot MarketPlace gift certificates.

So, what better way to spend a birthday than fighting robots in good company? Ram Robotics' Elizabeth McAmis made birthday cakes for Busted Nuts Robotics team members Mike Gellatly and Paul Grata, who were both celebrating a birthday at the event. The delicious creation resembled each of their bots, and the robot cakes quickly disappeared at the hands of the hungry competitors!

The night ended with many of the competitors driving over to the new store location of Robot Marketplace with owner Jim Smentowski to race R/C off-road trucks in his new indoor track. It turns out that driving robots did not prepare us for the havoc that was about to ensue! We raced the cars under a bridge, over jumps, and even onto a huge ramp with a spike strip.

FIGURE 10. Post-competition birthday celebration at Jim Smentowski's Hobby Superstore.

While driving the cars into walls, into other cars, and rolling them over more than a few times, spectators from the Sarasota Square Mall gathered around to watch. Luckily for us, the Hobby Superstore had some savvy mechanics on hand! **SV**

STEER WINNING ROBOTS WITHOUT SERVOS!

Perform proportional speed, direction, and steering with only two Radio/Control channels for vehicles using two separate brush-type electric motors mounted right and left with our **mixing RDFR dual speed control**. Used in many successful competitive robots. Single joystick operation: up goes straight ahead, down is reverse. Pure right or left twirls vehicle as motors turn opposite directions. In between stick positions completely proportional. Plugs in like a servo to your Futaba, JR, Hitec, or similar radio. Compatible with gyro steering stabilization. Various volt and amp sizes available. The RDFR47E 55V 75A per motor unit pictured above.

www.vantec.com

VANTEC Order at (800) 882-6832

Build the Kronos Flyer

Part 6: APM 2.5

by Michael Simpson

www.servomagazine.com/index.php?/magazine/article/april2013_Simpson
Discuss this article in the SERVO Magazine forums at http://forum.servomagazine.com

This series was originally supposed to run for only five months, and while last month's article was intended to be the last, I received a ton of requests to do a write-up of the APM 2.5 controller on the Kronos Flyer. So — by popular demand — here it is.

APM 2.5

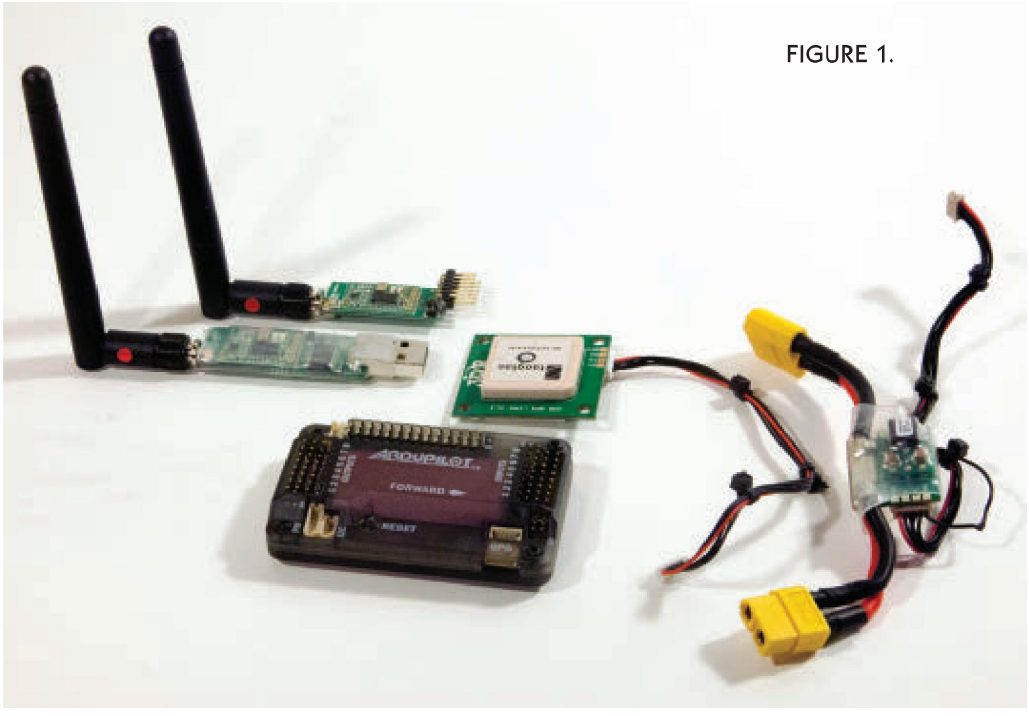


FIGURE 1.

Last month, I touched on the APM 2.5. The system shown in **Figure 1** consists of the APM 2.5+, GPS, and 915 MHz two-way telemetry radios. In my case, I upgraded to the better uBlox GPS unit.

You can purchase the main unit in both side entry or top entry for your hookup. I prefer the top entry as it allows for a smaller footprint on the quad platform.

While shopping for the APM, you may see the nomenclature APM 2.5+. This just refers to the APM 2.5, APM 2.5 case, GPS, and power module. In my case, I purchased the APM 2.5+ set and the telemetry radios.

New Firmware

One of the issues I had with the APM 2.5 system is that the stability of the NAZA blows the APM out of the water. It turns out there is a reason for this. The firmware in the APM was not fully utilizing the built-in accelerometers which effectively made the controller a three-axis system.

This has all changed.

With the release of the ArduCopter 2.9.1, the APM 2.5+ is now fully utilizing the accelerometers and is now a full six-axis controller. Does it make a difference? Yes, it does.

While the APM is still not on the same level as the NAZA, it is much easier to fly and — in my opinion — is a good alternative to the NAZA.

In the past, I had stated that the APM 2.5 was a controller that promised a lot but delivered very little. I can honestly say that now, I will be utilizing the APM 2.5 on many of my other craft.

Why would I do this when the NAZA is more stable, easier to install, and requires little or no tuning to get it to fly?

There are two reasons. I can get an APM 2.5 with GPS for as little as $179. The NAZA with GPS will run you $399. Is the NAZA twice as stable as the APM 2.5? Probably. However, the cost is not the only reason. It's all about those promises that the APM 2.5 makes.

For $85, I can add a two-way radio system that can be used to send information back to my laptop. This will allow me to not only log various settings on the craft in real time, but also to add the ability to track the craft on a map.

The Mission Planner (shown in **Figure 2**) is an application that runs on Windows. It's the program you use to configure your APM 2.5 and to update your APM's firmware.

In addition, you can track your GPS location with the two-way telemetry radio and send commands to your craft. You can even fly your craft with a joystick.

Now, that's a lot of promises. No other system offers this much control for so little cost.

FYI ... that is my house and truck in the satellite view shown in **Figure 2**. I am sitting in my basement lab which is where the craft is showing up on the map.

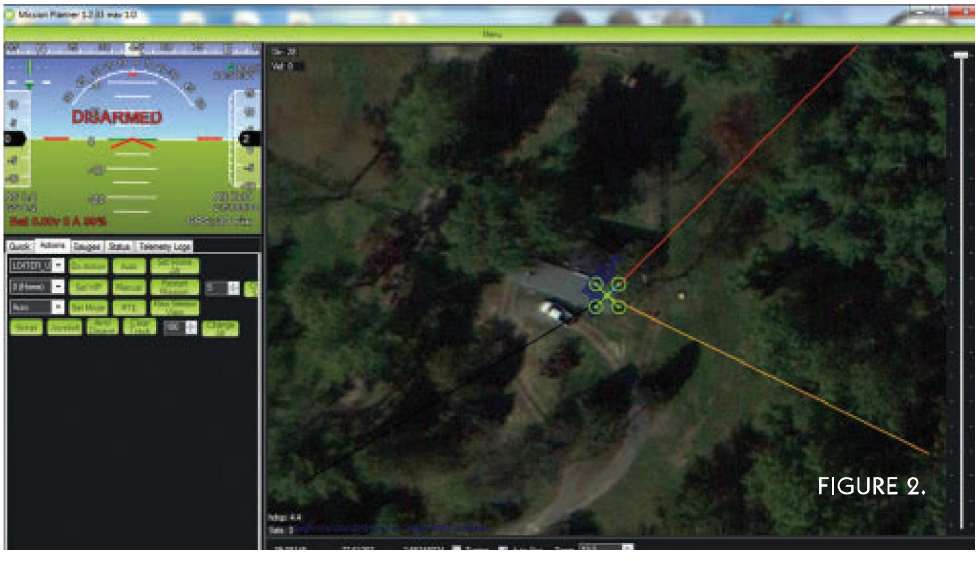


FIGURE 2.

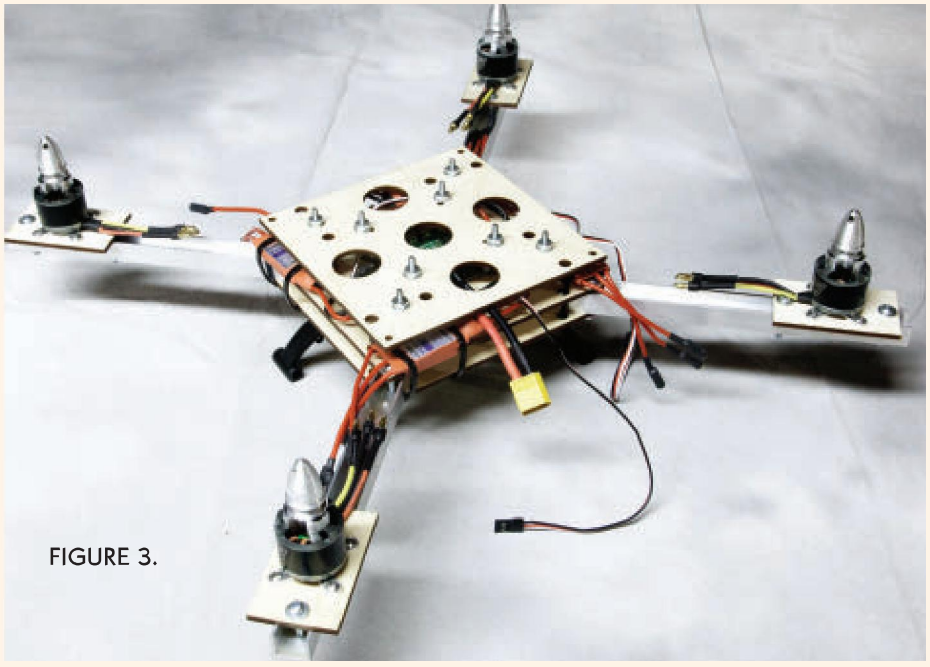


FIGURE 3.

Good Vibrations

Fully utilizing the accelerometers does come at a cost. Accelerometers are very sensitive to the propeller vibration in a multi-rotor craft. While these vibrations are handled by the NAZA enclosure, the same can't be said for the APM 2.5.

You must add vibration control to your craft before you can use the new firmware. Failure to do so could lead to your craft flying out of control.

You might be asking, what is vibration control? This can be as simple as adding foam rubber under the controller or as complex as an isolation platform on your craft.

I prefer the isolation platform because it allows me to add other things without having to worry about the

Boom Extensions

I talked about boom extensions last month, and have come up with several extensions that will fit the Kronos Flyer booms. These will allow you to use various size motors, and mount the motors on the top or bottom of the extensions as shown in **Figure 5**. While boom extensions are not required, they will allow you to use larger props and motors. They also look cool.

Landing Feet

Because I wanted to move the battery off the bottom of the craft, there is no longer a need for the helicopter style landing gear on the Kronos Flyer. As a replacement, I designed a set of small feet and boom covers shown in **Figure 6**. Again, these are optional, but will keep the belly of your craft off the ground for take-offs and landings. They also provide a good mechanism for attaching foam for a more cushioned set of landing pads.

Isolation Platform

The real meat of the KFE modular system is the redesigned isolation platform

FIGURE 5.

FIGURE 6.

vibrations caused by the propellers.

As I started to work with the APM 2.5 and its new firmware, I came up with new design enhancements for the Kronos Flyer.

KFE is Born

The original design of the Kronos Flyer was a real challenge. I wanted a frame that could be built with basic materials and tools. This way, anyone could recreate what I call "the reference frame" shown in **Figure 3**.

I also wanted to be able to add enhancements or upgrades to the designs that were a bit more sophisticated. These upgrades would take the basic frame to a whole new level. I call the enhanced craft the KFE for "Kronos Flyer Enhanced."

It's a modular design, so that only one or all of the enhancements can be added to the basic Kronos Flyer.

The frame in **Figure 4** shows enhancements to three areas. Let's take a closer look at these.

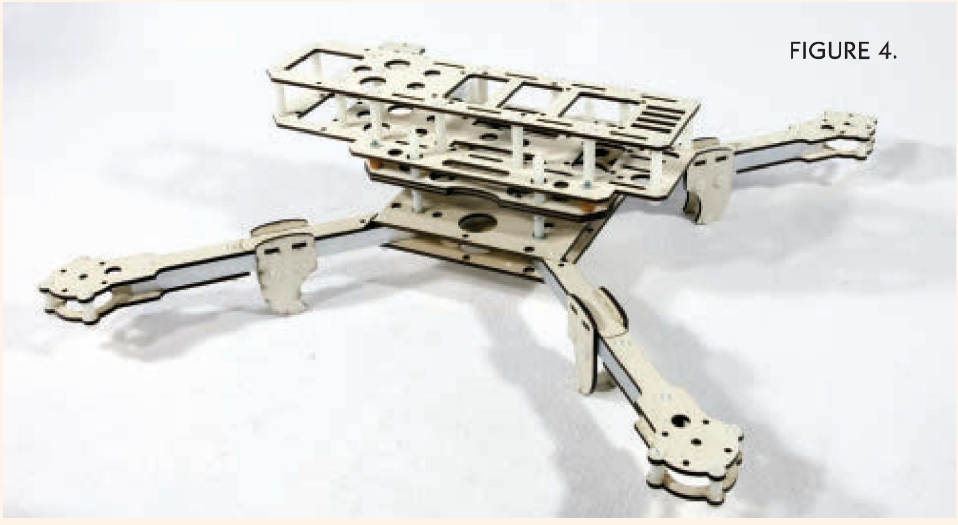


FIGURE 4.

shown in **Figure 7** which was designed specifically for the APM 2.5+ system.

One of the problems with many other platforms is the lack of room for mounting all your components. This platform has room for mounting the APM 2.5+, GPS, telemetry radio, FPV radio, FPV camera, and GoPro camera — all on the inside of the isolation platform. This leaves the top available for mounting any size battery — everything from the three-cell 2,200 mAh battery shown in **Figure 8**, to the four-cell 5,000 mAh battery shown in **Figure 9**.

You might be asking, why did you mount the battery on top of the isolation platform? The answer is simple: It adds to the mass of the isolation platform. This extra mass keeps even more of the vibrations from transferring from the main platform to the isolation platform.

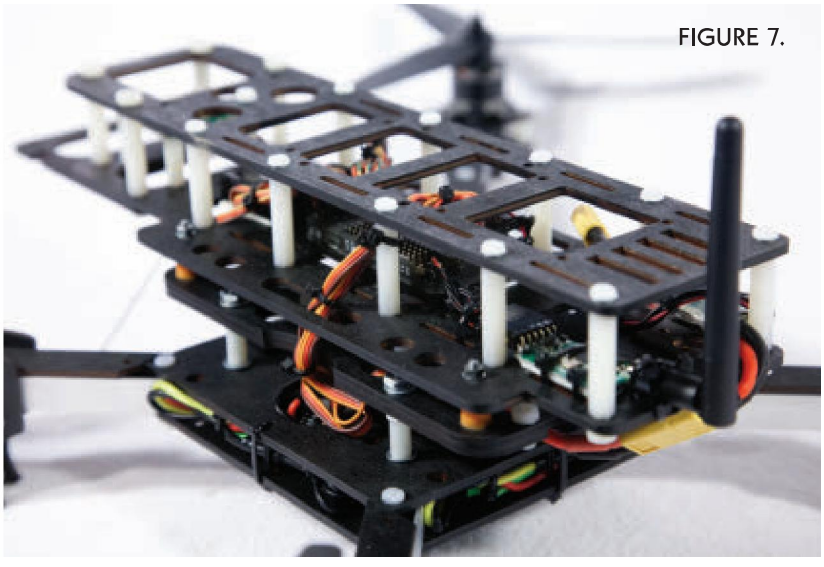


FIGURE 7.

The isolation platform is made from 1/8" Baltic birch plywood. I had thought of using G10 fiberglass, but the ply is lighter, stiffer, and easier to work with than G10. The isolation platform has a stiffened support platform that attaches directly to the original Kronos Flyer platform.

The support platform is tied to the upper isolated platform via four isolation bobbins. These bobbins, plywood, and added mass make for a vibration-free platform.

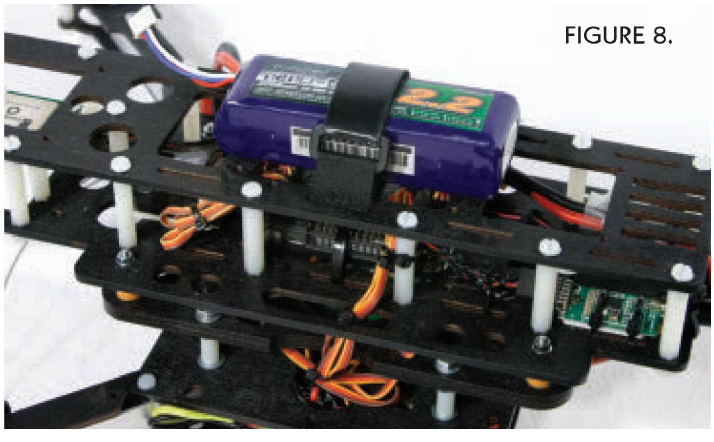


FIGURE 8.

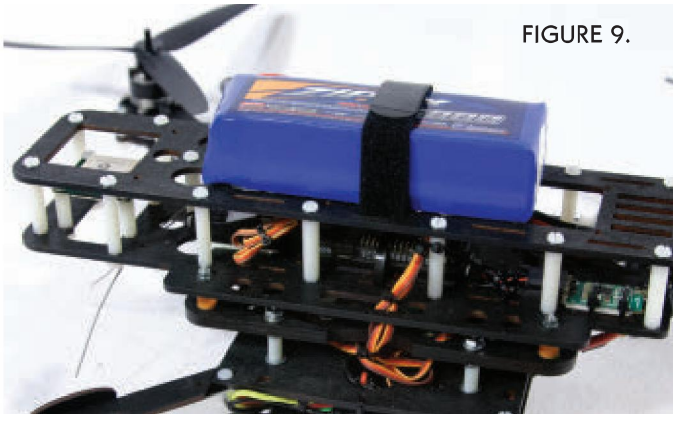
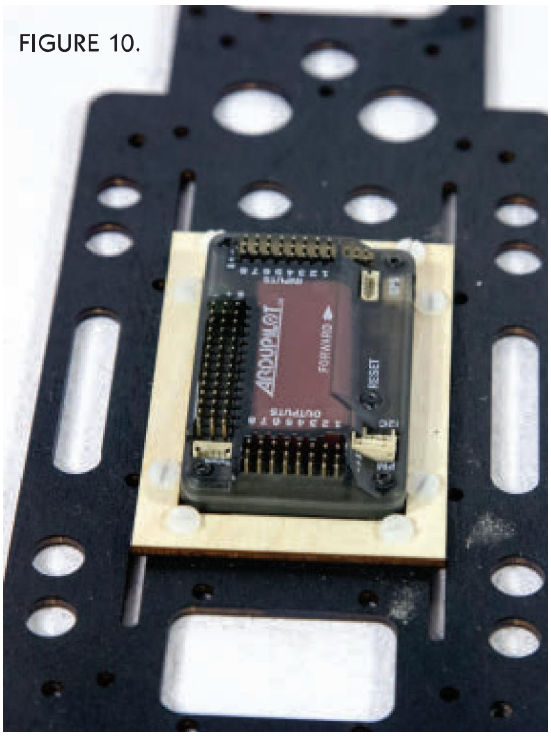


FIGURE 9.

FIGURE 10.



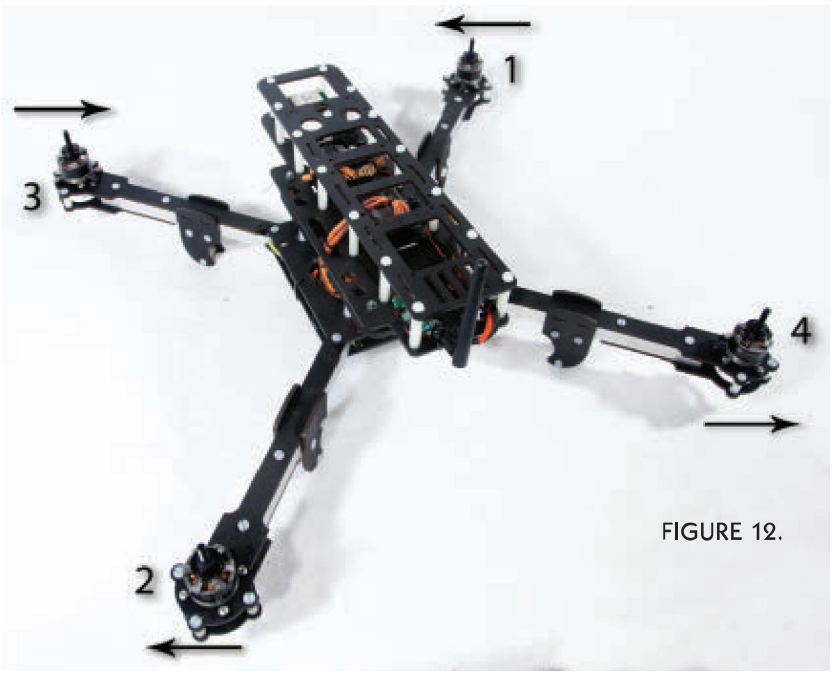


FIGURE 12.

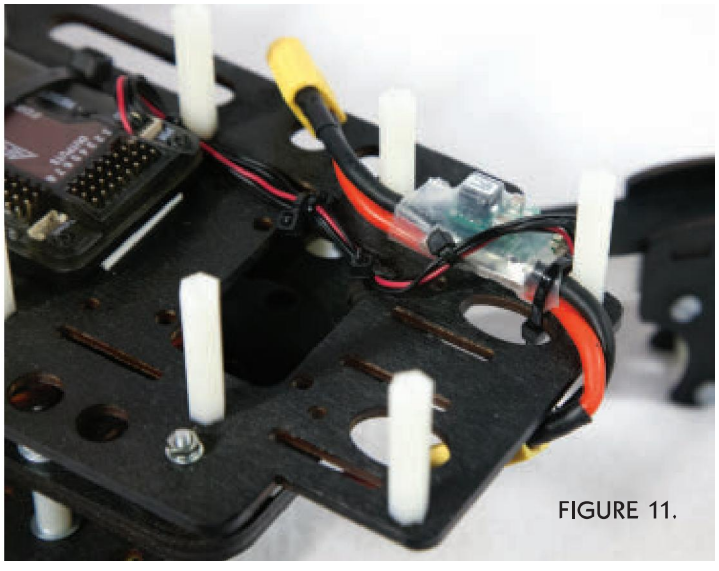


FIGURE 11.

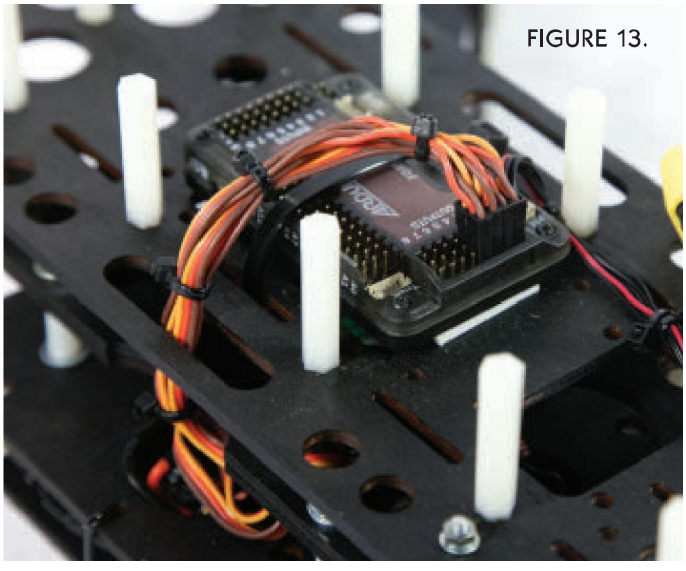


FIGURE 13.

APM 2.5+ Hookup

The APM isolation platform comes with an APM 2.5 alignment guide as shown in **Figure 10**. This guide will enable you to perfectly align the APM to the center of the craft. Next, the power module is attached to the rear of the platform as shown in **Figure 11**.

The power distribution side of the module is routed to the rear and attached to the battery connector on the power distribution board of your craft. The battery connector on the module is routed up and to the side to make access easy for attaching your battery.

In a nutshell, the power module connects between your battery and the APM. A small connector is then attached to the APM, thus providing power to the APM and battery sensor telemetry.

The next step is to attach your ESCs to the APM, but you need to set the motor direction before connecting them. Refer back to Part 4 in this series for instructions on how to set the motor directions. The rotation direction for the motors is shown in **Figure 12**.

Once your motors are set in the correct direction, it's time to connect them to the APM as shown in **Figure 13**. Route the cables through one of the holes in the support platform, then through the slot in the lower portion of the isolation platform. Using **Figure 12** as a guide, connect each ESC to the appropriate channel on the APM.

Mount your receiver in the space in front of the APM as shown in **Figure 14**. Connect each channel on the radio to each of the input channels on the APM. Tie wrap your cable into a neat bundle.

While the APM can be flown with a six-channel radio, I recommend using eight channels. This will give you ample control for more functionality.

Next, you will install the GPS unit. First, install a set of small standoffs, then attach the GPS to the standoffs as shown in **Figure 15**. Connect the GPS cable to the APM connector marked GPS. For this GPS unit, it's the GPS connector on the side of the APM.

The last APM component to connect is the telemetry radio. The best way to install the radio is to place two pieces of thick, double-sided foam tape to the bottom of the radio.

Stick it to the rear of the platform as shown in **Figure 16**. Use some tie wraps to help hold the radio in place in case the foam tape fails.

Since the antenna hangs off the back, I added a couple of tie wraps to help secure it, as well. Plug the connector into the radio as shown. The end of the connector with the missing pin is oriented so that it is facing the power module.

Plug the other end of the cable into the APM connector marked Telem.

The top is then added to the isolation platform and secured in place with screws.

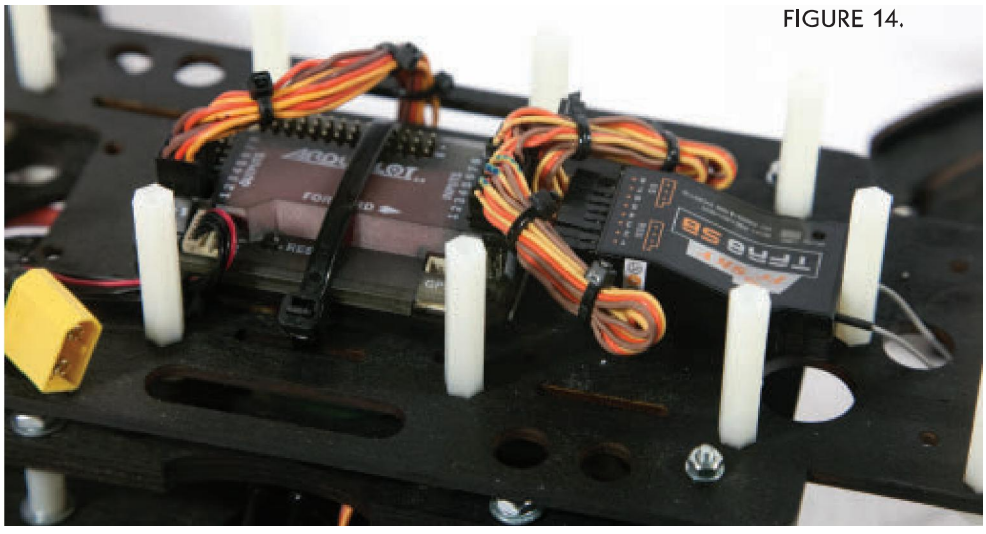


FIGURE 14.

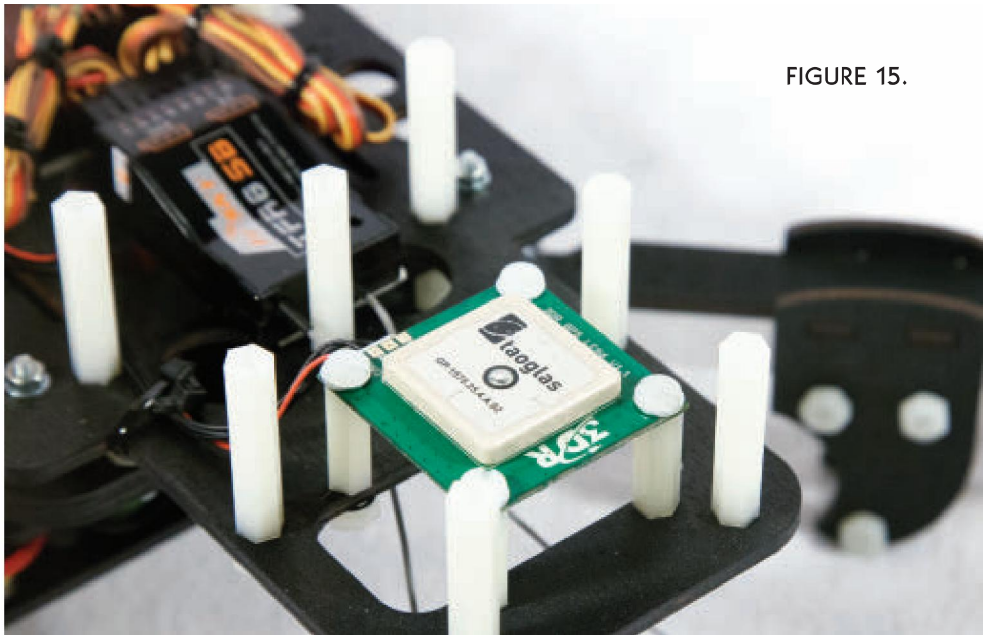


FIGURE 15.

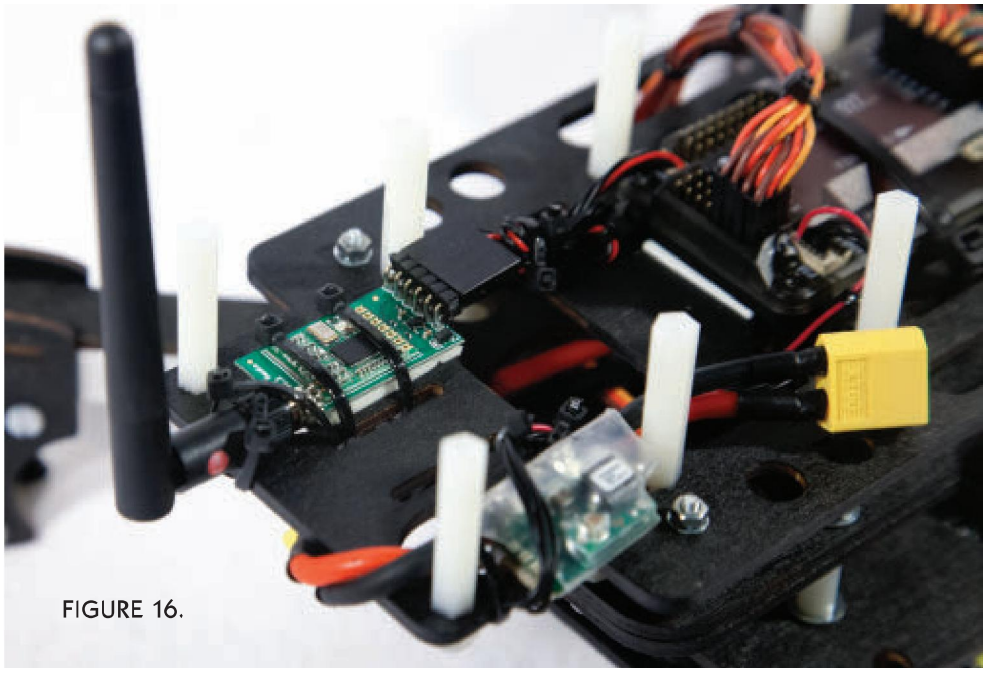


FIGURE 16.

APM 2.5 Software

The first thing you need to do is download and install the latest Mission Planner. You can get it at **http://code.google.com/p/ardupilot-mega/downloads/list**.

The one I downloaded was the MissionPlanner-1.2.33.msi. This one contains the ArduCopter 2.9.1 firmware. If you have an older version of the Mission Planner software, I recommend you uninstall it and install the 1.2.33 directly. I tried the built-in upgrade and it failed on all of my machines.

You can get a full walk-through for the Mission Planner software at **http://code.google.com/p/arducopter/wiki/AC2Installation**.

Complete ArduCopter instructions can be found at **http://code.google.com/p/arducopter/wiki/ArduCopter**.

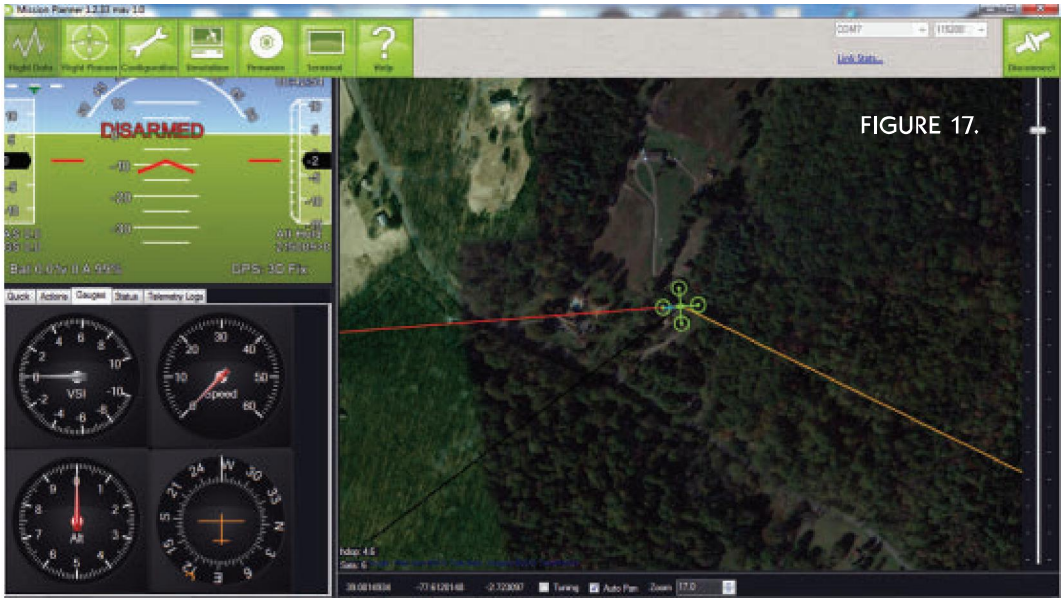


FIGURE 17.

Connecting to APM 2.5

Connect the ArduPilot to your PC with the included USB cable. This will create a new COM port on your machine. Go to the device manager and look in the Ports section. Mine was called Arduino Mega 2500 (COM7). Write down this port name. Note: You don't need to plug your battery into your craft.

Start the Mission Planner software. In the upper righthand corner of the Mission Planner, select the COM port that was installed when you plugged your APM into the USB on your computer. It should default to 115200 speed. You may be able to select the Auto option, as well. If you have other COM ports on your machine, this option can be problematic. Hit the Connect button.

The Mission Planner will go through some modules, and eventually connect to your APM 2.5. You may or may not see telemetry data in the Flight Data tab. It all depends on your firmware status. At this point, you can disconnect from your APM by hitting the Disconnect button.

Next, you need to upload the latest firmware to the APM. To do this, select the Firmware Tab on the Mission Planner. Click on the ArduCopter V2.9.1 Quad and start the download/update process. This will load the very latest quadcopter firmware into your APM.

Once the firmware has been updated into your APM, go back and hit the Connect button again. Once connected, you should see data in the Head Up display; if you have a GPS lock, you should see your location on the map as shown in **Figure 17**.

Configure the APM 2.5

Basic Configuration

Go to the Configuration tab at the top of the screen and select the ArduCopter Level tab on the left side. This will bring up an additional screen that will allow you to select a + or X configuration. Select the X configuration if it is not already selected.

Radio Calibration

If you have not bound your transmitter to your receiver, you need to do it now. While still in the Configuration tab, select the Radio Calibration side tab. With your radio turned on, refer to **Table 1** and make sure each transmitter channel is moving the bars on the calibration screen in the correct direction. If they are not, you will need to reverse them on your radio.

Once you are satisfied the directions are correct, hit the Calibration button and follow the on-screen instructions.

Channel	Radio Description	Direction 1	Direction 2
1	Ailerons	Left = Roll Low	Right = Roll High
2	Elevator	Up = Pitch Low	Down = Pitch High
3	Throttle	Down = Throttle Low	Up = Throttle High
4	Rudder	Right = Yaw High	Left = Yaw Low
5	Three-Position Switch 1	Radio 5 - Three-Position	Radio 5 - Three-Position
6	Knob	CW = Radio 6 High	CCW = Radio 6 Low
7	Three-Position Switch 2	Radio 7 - Three-Position	Radio 7 - Three-Position
8	Knob	CW = Radio 8 High	CCW = Radio 8 Low

Table 1.

Flight Modes

With your radio calibrated, you can now set your flight modes. Select the Flight Modes tab, then move your channel 5 switch. If you have a radio with a three-position switch, then you will be able to select one of three flight modes. If you only have a two-position switch, then only one of two can be selected. As you change the flight mode, you should see the appropriate mode field highlighted. On my three-position switch, modes 1, 4, and 6 can be selected. For them, I chose Stabilize, Alt Hold, and Loiter as shown in **Figure 18**. Save the modes once you have them set the way you want.

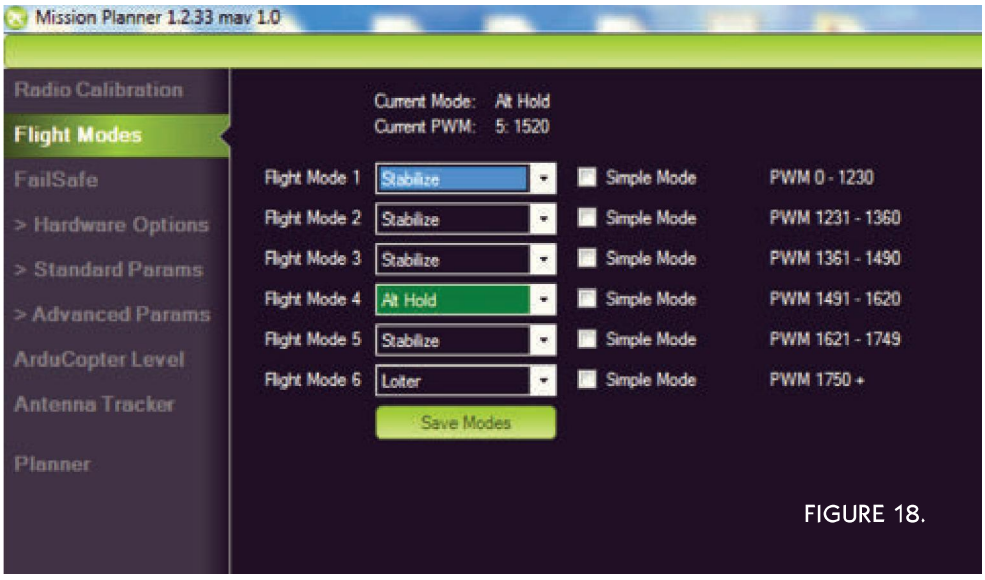


FIGURE 18.

You need to select the modes for your craft. Use **Table 2** as a basic guide for the three modes I just mentioned. Refer to the ArduCopter documentation for descriptions of all the modes.

Mode	Description
Stabilize	This mode keeps the craft upright and somewhat steady. You manually control altitude with the throttle.
AltHold	This mode is the same as Stabilize mode but will hold its altitude as long as the throttle stays in the center position. Raising and lowering the throttle will allow you to move to a new altitude.
Loiter	This mode will attempt to use the GPS to hold the craft in one position. How successful this is depends on the number of satellites you are locked onto and how much interference the GPS is getting.

Table 2.

Hardware Options

Select the Hardware Options side tab. Make sure the Compass is enabled. This section is where you can select other hardware options to add further control of your craft. The Sonar sensor adds more precise control of altitude when you are close to the ground. The Airspeed sensor adds a true air speed option to your telemetry. The Optical Flow sensor allows your craft to track ground targets. Please be advised, because I am not sure on the actual support of these sensors. I'm pretty sure the Sonar sensor is working with the current firmware.

Accelerometer Calibration

Go back and select the ArduCopter Level side tab. Hit the Calibrate Accel tab and follow the on-screen instructions. It may be helpful to use a table or other large object to help steady the craft as you move it from position to position.

ESC Calibration

While the APM instructions state that you can use the stand-alone method (the one I used in Part 4), it does not work. I found the following pass-through method works best with the APM 2.5. Note that your props should not be connected.

Step 1
Disconnect the USB and battery from your craft.

Step 2
Turn on your transmitter. Once it is on, set the throttle at its max.

Step 3
Connect the battery to your craft. After the APM boots, you will see the main LED lights cycle through yellow, red, and blue. The APM is telling you it is ready to place the APM in pass-through mode.

Step 4
With the throttle still set to full, remove the battery from your craft and then reconnect it. The ESCs should play a tune, pause, then do a single beep. When this happens, lower your throttle all the way and the ESCs should beep again. The ESCs are now calibrated.

Step 5
Move the throttle up a little and make sure all the motors start at the same point.

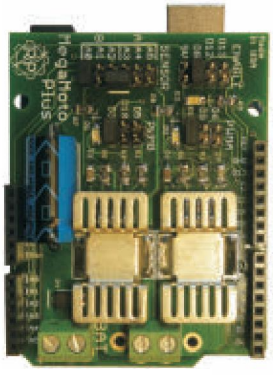
Ready for Flight

You have now done the minimum configuration required to fly your APM 2.5 KFE. Be sure to balance your props, then mount the following props in the following positions:

Motor 1 = Left hand prop
Motor 2 = Left hand prop
Motor 3 = Right hand prop
Motor 4 = Right hand prop

For your first flight, make sure you have plenty of room. Don't do it indoors.

Extreme Motor Speed Control!



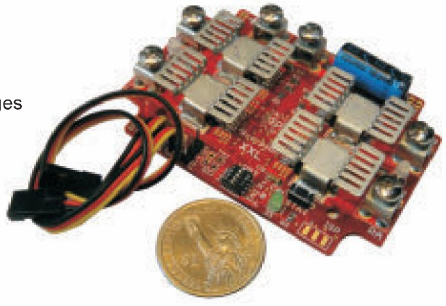
MegaMoto Plus Shield

High-current motor control for Arduino™

- 7V-28V - H-bridge or Dual Half-bridge
- **30A Continuous** with fan - **40A peak**
- Current sensor outputs can be sent to any analog pin
- **Stack** up to three units on one Uno or compatible
- Independent half-bridges can control brushless/steppers too!
- Over temp and over current protection

Scorpion XXL

- 7V - 28V
- Dual 40A+ Peak H-Bridges
- Current/Temp limiting
- R/C inputs w/mixing
- Optional enclosure
- 3.25" x 2.25" x .5"



NEW!

Vyper

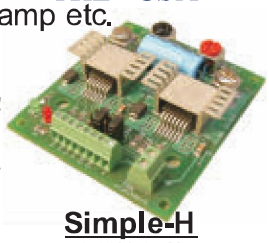
- **Single 120A+ H-Bridge**
- 8V - 42V
- R/C inputs w/mixing (2 units req.)
- Analog pot input
- Supports 8 AWG wires!
- Optional enclosure
- 2.85" x 2.25" x 1.75"

MADE IN THE USA

H-bridges: Use with Dalf or Stamp etc.

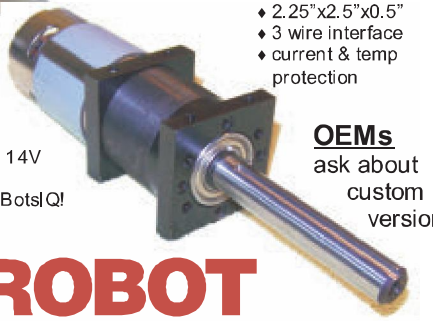
OSMC

- Monster power!
- 14-50V **160A!**
- 3.15"x4.5"x1.5"
- 3 wire interface



Simple-H

- 6-28V **25A!**
- 2.25"x2.5"x0.5"
- 3 wire interface
- current & temp protection



Magnum775

- RS-775 motor
- planetary gearbox
- 20:1 ratio - 700 rpm @ 14V
- Nearly 700W!
- Build something - rule Bots!Q!

OEMs ask about custom versions

ROBOT POWER™

www.robotpower.com

Phone: 360-515-0691 • sales@robotpower.com

Conclusion

The configuration I have shown you here only scratches the surface of what the AMP 2.5 KFE is capable of. I don't recommend you go out and start running waypoints or doing circle maneuvers until you have several flights under your belt. This will also let you work out any kinks in your craft.

To date, I have only flown a couple of waypoints successfully and that is because I worked the craft with the radio and someone else operated the laptop. I'm sure with time I will be able to do both, once I get a few of my own kinks out.

Be sure to check out the Kronos Robotics website for KFE parts and more detailed instructions for assembly. I plan on offering various upgrade options, as well as a complete frame kit.

As always, be sure to post your questions in the *SERVO Magazine* forums at **http://forum.nutsvolts.com/viewtopic.php?f=49&t=16866**. SV

Next Month: 3D Printers

Starting next month, I will begin a new series on 3D printers. I will take you through the build process of the two printers shown in **Figure A**.

The Rostock MAX (on the left) is probably one of the largest 3D printer kits you can put together. It's based on a Delta Robot design and is as much fun to watch as it is to print.

The PrintrBot Jr (on the right) is probably the smallest. This little guy is so small I can place it just about anywhere. While not as sophisticated as the MAX, it's still a lot of fun to tinker with.

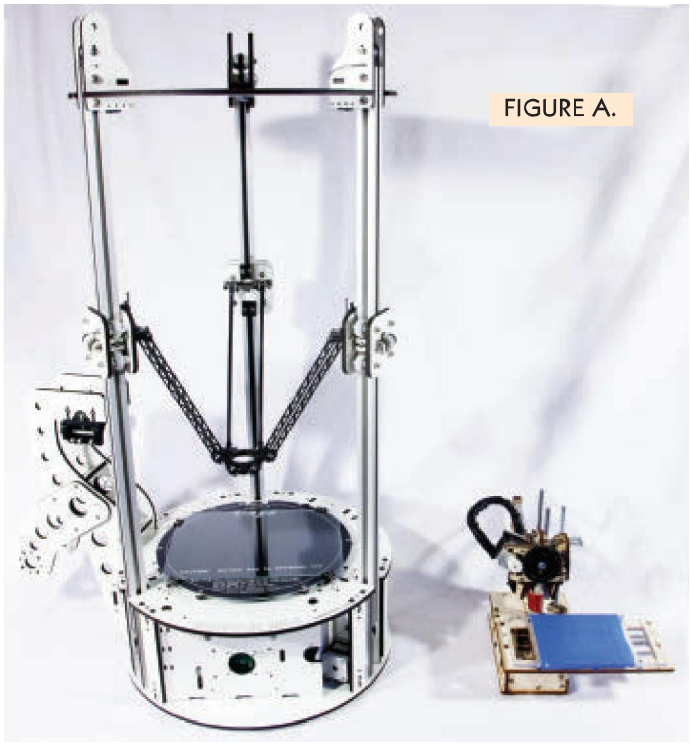


FIGURE A.

Maker Faire®

8th Annual Bay Area
SAN MATEO EVENT CENTER

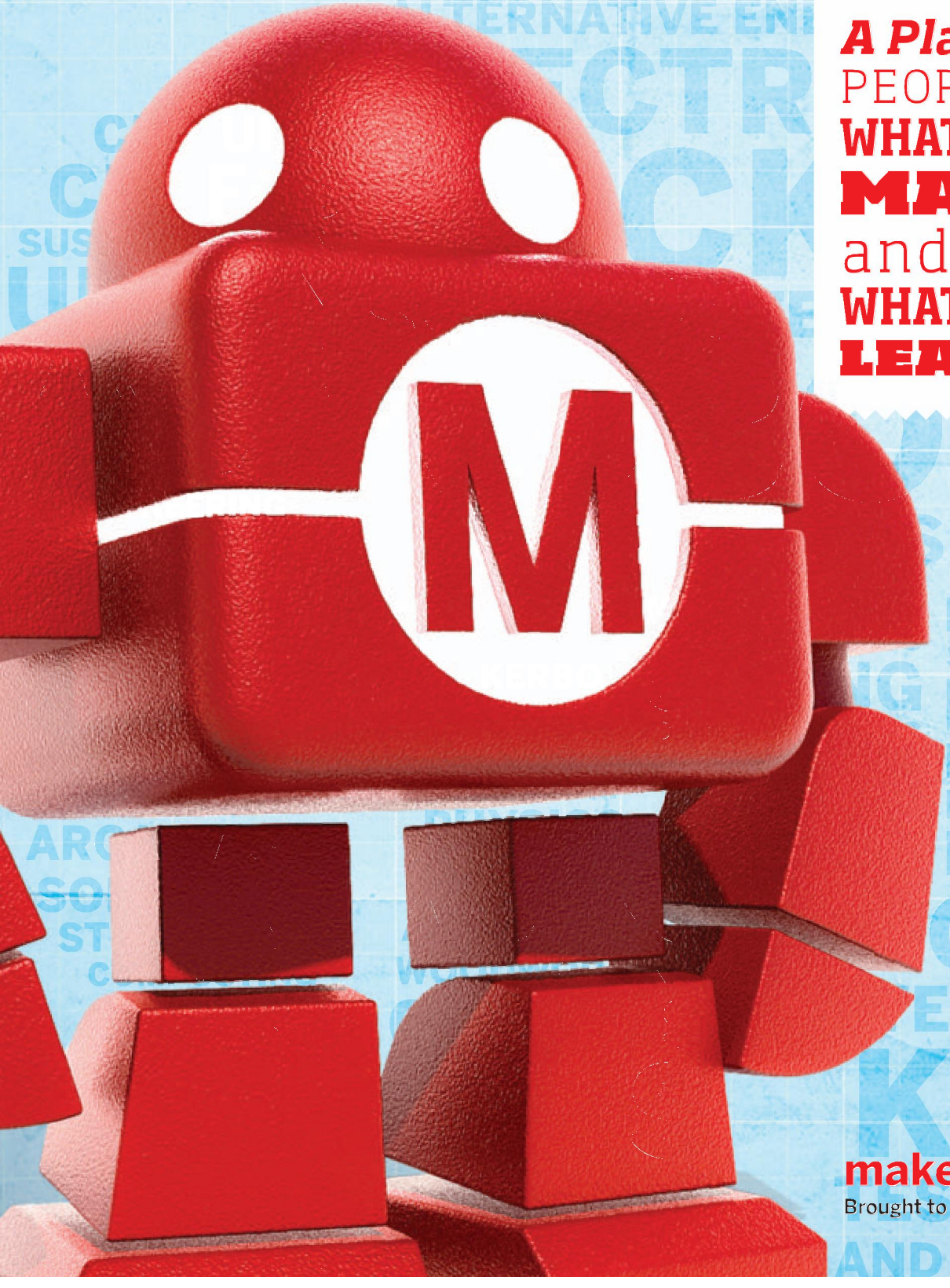
MAY 18–19

★★★

A Family-Friendly
SHOWCASE OF
INVENTION
CREATIVITY &
RESOURCEFULNESS

★★★

A Place Where
PEOPLE SHOW
WHAT THEY ARE
MAKING
and SHARE
WHAT THEY ARE
LEARNING



THE GREATEST SHOW (and TELL) ON EARTH

makerfaire.com

Brought to you by **MAKE** magazine

Last month, we saw how a robot could find and identify a ball and goal, allowing it to engage in a simplified version of soccer. This month, we will examine how the principles of identification used with the soccer robot can allow a mobile robot to find and retrieve objects with a simple robot arm and then move them to a desired destination.

Robotic Sensors

Part 2: Their Use With Robotic Arms

by John Blankenship and Samuel Mishal

www.servomagazine.com/index.php?/magazine/article/april2013_Blankenship
Discuss this article in the SERVO Magazine forums at http://forum.servomagazine.com

As mentioned, last month we used a simplified soccer-playing robot to demonstrate that a robot's ability to cope with its environment can be improved by giving it a rudimentary ability to identify objects of importance. The ball trap on the soccer robot was essentially a non-motorized gripper, and the robot itself could be thought of as a robotic arm with one degree of freedom. Because of these similarities, it makes sense to expand on principles discussed last month by applying them to a more capable robotic arm.

Before we begin though, we need to examine robotic arms themselves. If you search the 'net for *robotic arm images*, you will find hundreds of arms that are suitable for the robot hobbyist. In almost every case though, the arms have a major flaw — the arms and grippers have little — if any — sensory capability. What's worse is that most arms have not even been designed to allow sensors to be easily added. This limits the arm's activity to pick-and-place maneuvers where the objects to be manipulated are assumed to be found at specific positions, sometimes with specific orientations. Industrial assembly lines can often make such assumptions, but a hobby arm — like most hobby robots — needs sensors to properly interact with its environment.

The lack of sensors may seem acceptable at first, because most low cost arms utilize servomotors to power their joints, allowing the arm to be positioned without the need for sensory feedback. This is not unreasonable as long as maximum loads are not exceeded, plus it has significant cost advantages. High-end digital servomotors — such as Dynamixels — have built-in sensory capabilities that can provide information about the position and load associated with an arm joint. However, this does not eliminate the need for gripper-mounted sensors that can be used to guide and control the arm's movements.

The arm's movement itself brings up another important point: Nearly all robotic arms have the motors that control the joints mounted on the arm itself. This causes two problems. The arm's lifting ability is diminished by the weight of the motors, and the movement of some joints can affect the orientation of others.

Figure 1 demonstrates this point

by showing that joint angles are relative to the position of other joints. As you can see, when the shoulder joint is moved, the angles of the other two joints (when referenced to ground) change dramatically.

When arms are built this way, a significant amount of mathematics is required to determine the angle each joint should have in order to create a desired position and orientation of the gripper.

Furthermore, if you want to keep the gripper level during some maneuver, for example, the elbow and wrist joints have to constantly be manipulated during shoulder joint movements.

If the motors are moved off the arm — as in a man-sized arm we built for another project (see **Figure 2**) — the orientation of each joint becomes independent of the other joints, and we get the added benefit that the arm does not have to lift its own motors. All the joint motors are mounted behind the shoulder joint, and a belt system delivers power to each of the joints as shown in **Figure 3**.

FIGURE 1.

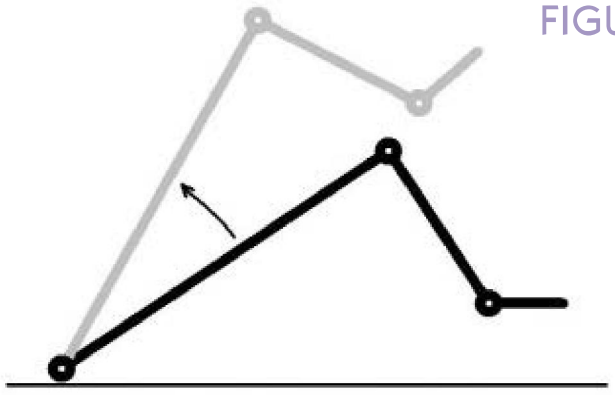


FIGURE 2.

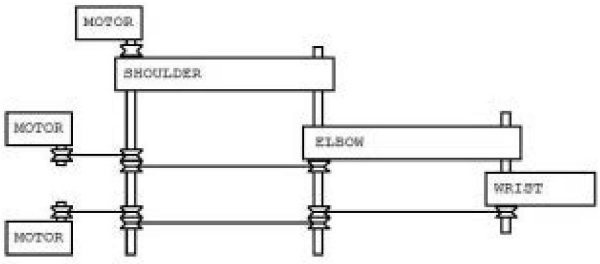
Easy Programming

If you program an arm built with the principles shown in **Figure 3**, you quickly appreciate its advantages. If you move the wrist joint to a level position, for example, it will remain level no matter how you move any of the other joints. It would be nice if some manufacturer would offer a similar design, but for now you will have to build your own.

We have provided details on the construction of this arm in the RROS manual (download at **www.RobotBASIC.com**), but we also understand that most hobbyists may not want to undertake a major construction project in order to experiment with the principles we are about to discuss.

For that reason, we also built a much smaller arm that utilizes many of the same concepts. It is much easier to build because it has only one moving joint. What may surprise you is that it retains much of the functionality of a far more complex version. The arm is small enough to fit on the RobotBASIC RB-9 chassis as shown in **Figure 4**.

FIGURE 3.



Constructing the Arm

Most of the structural components of the arm and gripper are made from various sizes of craft sticks, but you can also use Popsicle sticks and coffee stirs. **Figures 5** and **6** show alternate views to provide additional detail. You can also refer to Figure 7 of last month's article which offers a clearer view since it showed the arm before the wiring was added.

Since this arm is small and lightweight, the axles for the joints are just bolts with plastic spacers serving as bearings. The wrist joint is fixed to the forearm, and the forearm maintains the position established by the two pulleys mounted on the shoulder and elbow joints.

Nylon line wraps around the elbow pulley and then around the shoulder pulley, thus ensuring the two pulleys stay in sync. The line then connects to two bolts in the base, rendering both pulleys in a fixed orientation regardless of the shoulder's movement. This means that the shoulder joint can move at will, and yet the gripper remains level.

Construction of the gripper can be tricky. As you can see in **Figure 6**, the fingers are attached to the base of the hand with two struts on each side. Since these struts are equal in length, they form a parallelogram which maintains the orientation of the fingers. Actually, each strut is made from two sticks to increase stability.

The bolts at each end of the sticks serve as axles and should be used with nylon nuts so they will stay fixed without being overly tight. We added a cross-member that slides between the strut sticks for additional stability, but this is not necessary if the ends of the struts have more overlap than the base of the hand and fingers.

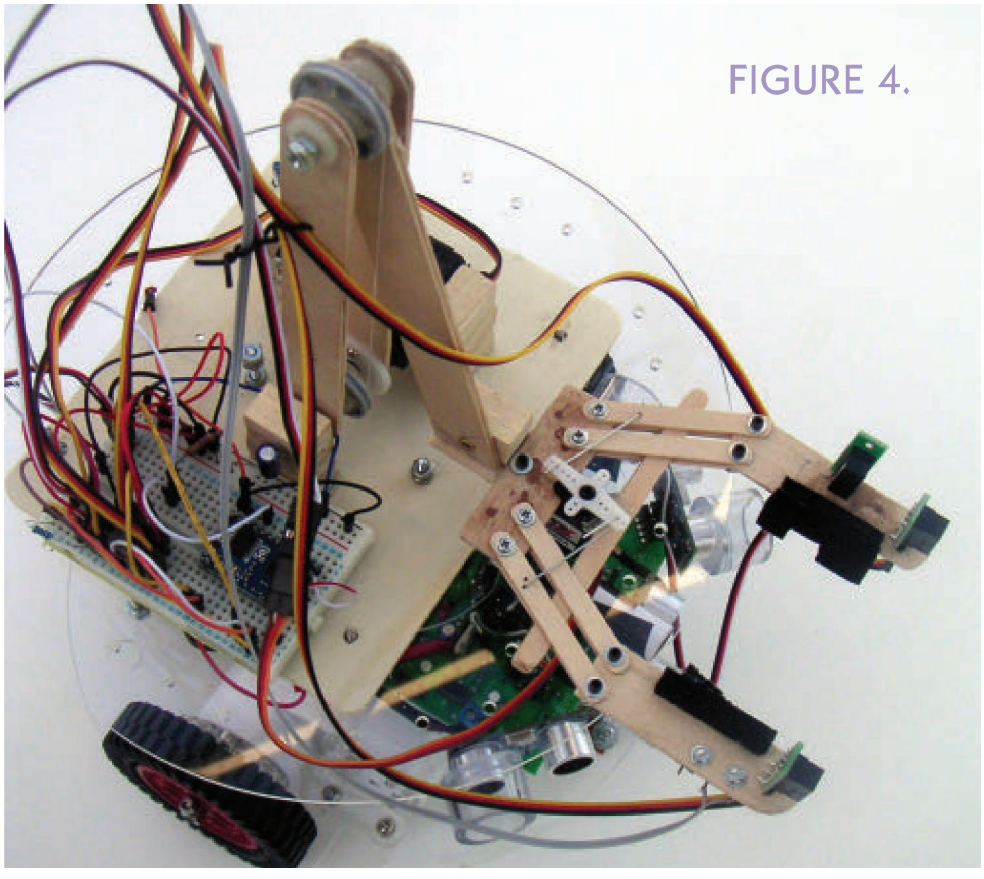
The nice thing about constructing with craft sticks is that they are very cheap and easy to work with, allowing you to experiment with numerous designs.

This one-motor arm is surprisingly agile. The gripper can be lowered to ground level to pick up objects and it can be raised so that objects can be placed on platforms or dropped into baskets. When the arm retracts, as in **Figure 4**, the gripper and any object in its grasp are out of the view of the robot's primary perimeter sensors, including the beacon detector.

A small servomotor opens and closes the gripper as shown in **Figure 6**. Notice that three sensors are mounted on the fingers just as they were on the ball trap used by the soccer robot last month. Foam rubber is mounted on the inside edges of the fingers to give a better grip.

Look carefully at the finger at the bottom of **Figure 6**. A small snap-action switch is mounted under the finger with foam glued to its lever (you can only see the lever). Reading the state of this switch makes it easy to determine when the gripper has closed on an object.

FIGURE 4.



Creating an Application

Our next step is to write a program that will allow the robot to find small objects and move them to a beacon in much the same manner as soccer robot. Most of the

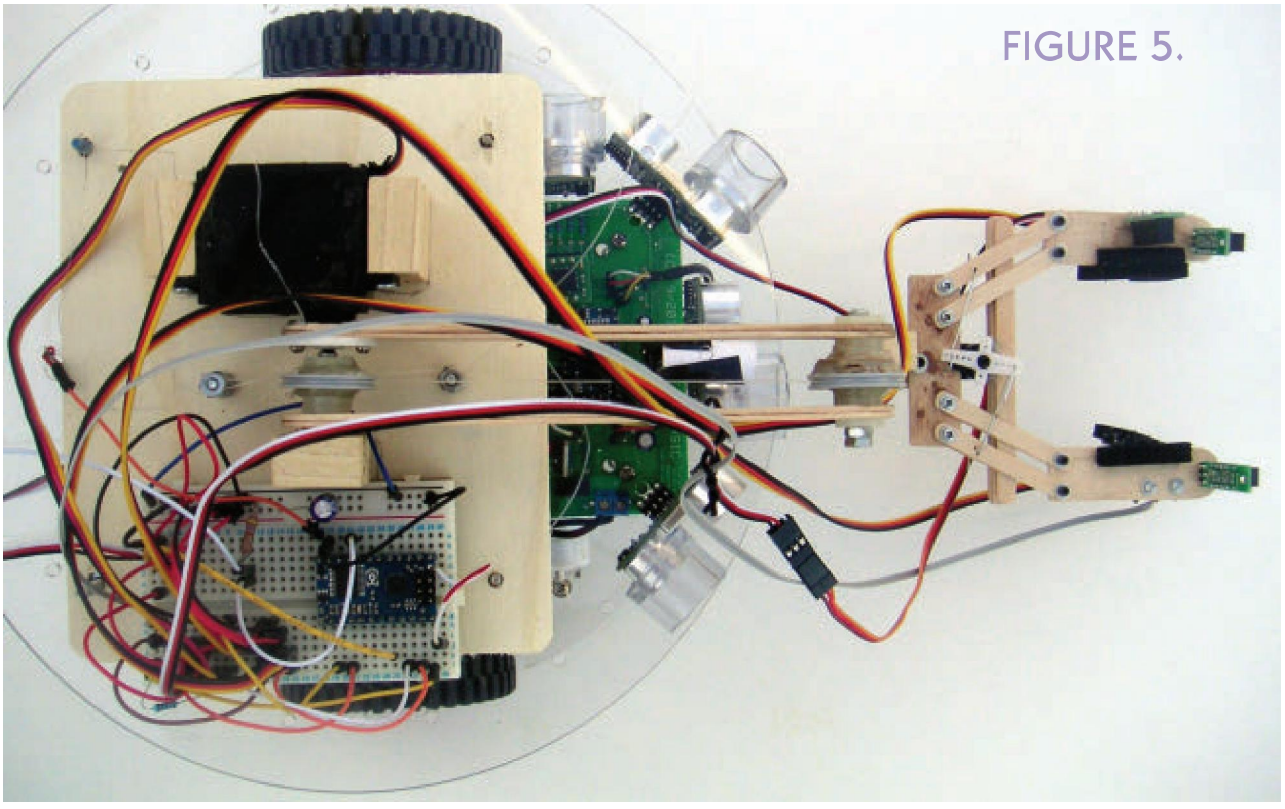


FIGURE 5.

software in last month's article can be reused since the arrangement of the gripper sensors (when the fingers of the gripper are open) is the same as the trap. This makes the programming much easier because we only have to add the details of the arm's movements.

Once it has been determined that an object is properly positioned in the gripper, the fingers can be closed until the snap action switch indicates that contact has been made. The amount of closure required of the servomotor serves as an indication to the size of the object being obtained.

Since objects can be distinguished by their size, the robot could be programmed, for example, to place large objects in one basket and small objects in another.

Since the shoulder joint is also a servomotor, the electronics for the arm can be any servomotor controller. Like last month, we are using the RobotBASIC RROS to control the robot, so we will also use it to control the arm. The RROS chip actually has an alternate identity. When a second RROS chip is connected to the main RROS, it can be instructed to become a servomotor controller. There are several advantages of using two RROS chips in this manner over using a conventional servo controller.

First, all the commands for servo control are automatically handled by the wireless link of the main RROS chip. Second, the RROS servo controller supplies additional analog and digital I/O lines that can be read with RobotBASIC commands. The analog inputs could be used with potentiometers to ensure that the arm joints are actually where they are suppose to be.

You could also use the analog lines to measure grip pressure or determine the color of an object held in the gripper (using appropriate photoresistors, etc.). The additional digital I/O lines are read as line sensors as in last month's program.

Use Any Microcontroller

Remember, you do not have to use the RROS system (or even RobotBASIC, for that matter) to experiment with the concepts discussed. The RobotBASIC code is easy to read and well commented, so porting it to other platforms and processors should not be difficult. If you are considering using the RROS though, know that complete details of its operation are available in the downloadable RROS manual.

Figure 7 shows the complete listing for the application program. Since the listing is commented and much of it utilizes the same algorithm used by the soccer robot, an extensive explanation is not necessary. The routines **GetObject** and **MoveToBeacon** are not shown since they are the same as last month. There are, however, a few things that should be mentioned. Let's look first at the main program.

The **GetObject** routine from last month allows the robot to find and move to an object. This time, we used inverted pill bottles as objects instead of the ball used last month. Once the bottle is in the hand (based

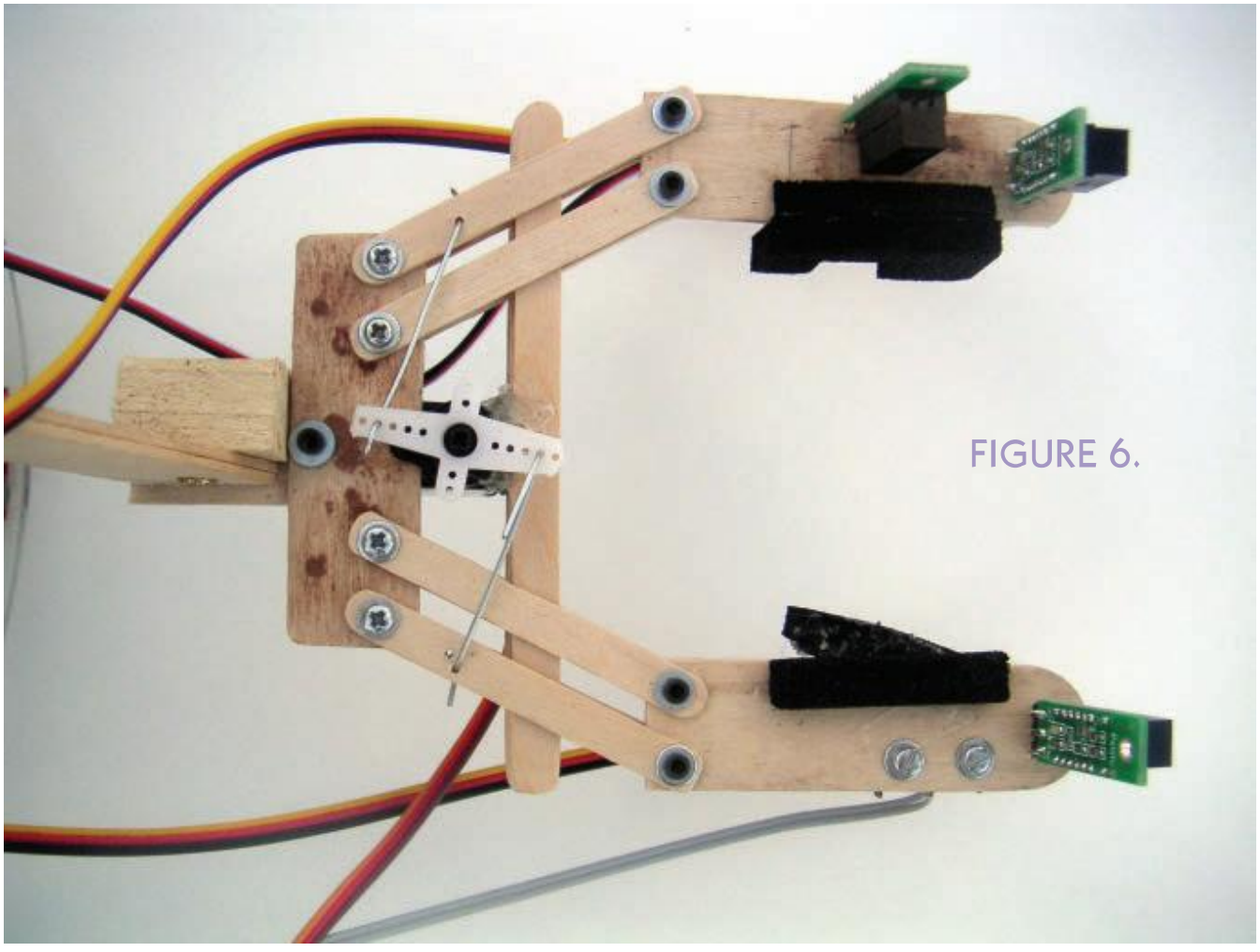
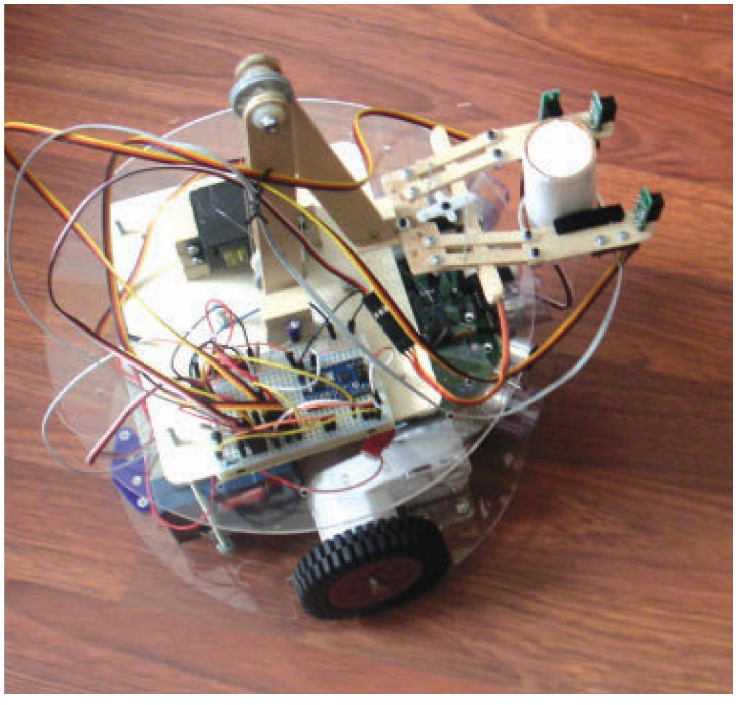


FIGURE 6.

FIGURE 8.



on the IR sensor that is pointed across the open fingers — just like last month), the hand is closed and the arm is raised to prevent it from interfering with other sensors. The **MoveToBeacon** subroutine from last month moves the arm toward the beacon.

The only change we made to this routine was to force the robot to stop when something was detected by the front perimeter sensor instead of the finger sensors used previously. This was necessary because the gripper is raised to prevent interference with the beacon detector.

When the beacon is reached, the arm is lowered and the object released. The arm is then raised back to its normal position.

There are several routines that initialize the arm and perform basic arm movements. Let's look first at the **ArmInit** routine. It begins by telling the main RROS chip that an external processor has been added, and that it is a second RROS chip used as an arm controller.

The next line tells the RROS to invert all four signals from the gripper sensors to indicate true when objects are detected. Next, the command **rSenseType** establishes that there are additional **rSense()** sensors

(normally only three are used). The remainder of the routine initializes the arm itself.

Most commands for controlling the servomotors require an index value to specify which joint parameter is being addressed. Once set, the index automatically increments each time a command is issued, making it very efficient to control an arm with multiple motors.

Since this arm has only one joint motor, the index is constantly set back to zero in the routines. Using an index of one allows control of the gripper's servomotor.

The RROS arm controller allows you to set a minimum and maximum point for each servomotor. Once set, the motor position (0-255) specifies a position between these limits. This means you have excellent resolution even if a particular servo only moves a small fraction of its typical 180º range.

This method also helps prevent a faulty program from moving a joint out of its preset range.

The demo program in **Figure 7** allows the robot to find an object that will fit into the hand and move it to a beacon as depicted by **Figure 8**. The variable **HandPosition** is set in the **HandClose** routine. Its value represents how far the fingers were closed before the finger-mounted switch indicated pressure from the object.

You can use this value to help identify the size of the object. If desired, the object could then be moved to various places by using different beacons to identify the possible destinations.

The RROS manual explains how RobotBASIC can identify 15 different beacons.

Conclusion

This article and the one from last month have shown how sensors can be used to locate and identify objects, so they can be manipulated by a mobile robot using a simple trap or arm to acquire the object and move it to a desired destination.

After experimenting with the principles demonstrated here, we suggest adding additional sensors (such as a color sensor) to allow your robot to identify and distinguish even more objects within its environment.

Since the arm can be raised to allow full use of the robot's perimeter sensors, this month's robot is ready for more complex experimentation. You could, for example, have the robot avoid objects in its path as it makes its way to the beacon. SV

FIGURE 7.

```
#include "RROScommands.bas"
#Include "InitializationRoutines.bas"
gosub InitRROScommands
gosub InitRB9Chassis

main:
  gosub ArmInit
  gosub GetObject // find something to pickup
  gosub HandClose
  gosub ArmUp
  gosub MoveToBeacon
  gosub ArmDown
  gosub HandOpen
  gosub ArmUp
end

ArmInit:
  rCommand(ExpansionSetup,ARM)
  rCommand(SetSenseInvMask,15)
  rSenseType 5 // allows more sensors
  rCommand(SetServoIndex,0) // index auto-incr.
  rCommand(SetServoSpeed,3) //arm
  rCommand(SetServoSpeed,3) //gripper
  // set minimums
  rCommand(SetServoIndex,0)
  rCommand(SetServoMin,150)
  rCommand(SetServoMin,110)
  // set maximums
  rCommand(SetServoIndex,0)
  rCommand(SetServoMax,45)
  rCommand(SetServoMax,35)
  gosub ArmHome //set home position & enable
return

ArmHome:
  rCommand(SetServoIndex,0)
  rCommand(SetServoPosition,0) // arm
  rCommand(SetServoPosition,0) // gripper
  rCommand(EnableServos,1)
  delay 2000 // allow time for movement
  // remove strain on the hand by
  // moving the fingers slightly
  rCommand(SetServoIndex,1)
  rCommand(SetServoPosition,20)
return

ArmDown:
  rCommand(SetServoIndex,0)
  rCommand(SetServoPosition,250)
  delay 2000
return

ArmUp:
  rCommand(SetServoIndex,0)
  rCommand(SetServoPosition,0)
  delay 2000
return

HandOpen:
  rCommand(SetServoIndex,1) // index for the hand
  rCommand(SetServoPosition,0)
  delay 2000
  rCommand(SetServoPosition,20)
return

HandClose:
  for HandPosition=0 to 250 step 3
    rCommand(SetServoIndex,1)
    rCommand(SetServoPosition,HandPosition)
    rforward 0
    if rSense()&8 then break // check for switch
  next
  //HandPosition now indicates size of object
return
```

Continued from page 19

app, and export of data.

Fully functional in-app demos of devices, monitoring, and programs are available with or without a connected XLN power supply. This is a convenient feature for pwrApp/XLN novices, and persons considering an XLN purchase.

B&K Precision's pwrApp offers full interface control in English, French, German, Portuguese, Spanish, and traditional Chinese via a simple selection menu. Future versions of pwrApp will offer control of additional select WLAN enabled B&K Precision power supplies. Available for immediate download via iTunes, pwrApp is free of charge to all users.

For further information, please contact:

B&K Precision

Website:
www.bkprecision.com

ROBOTSHOP WILL INTRODUCE...

SERVO ERECTOR SET V1.1

05.24.2013

GET READY TO CHANGE THE WORLD OF ROBOTICS

Lynxmotion

Imagine it. Build it. Control it.™

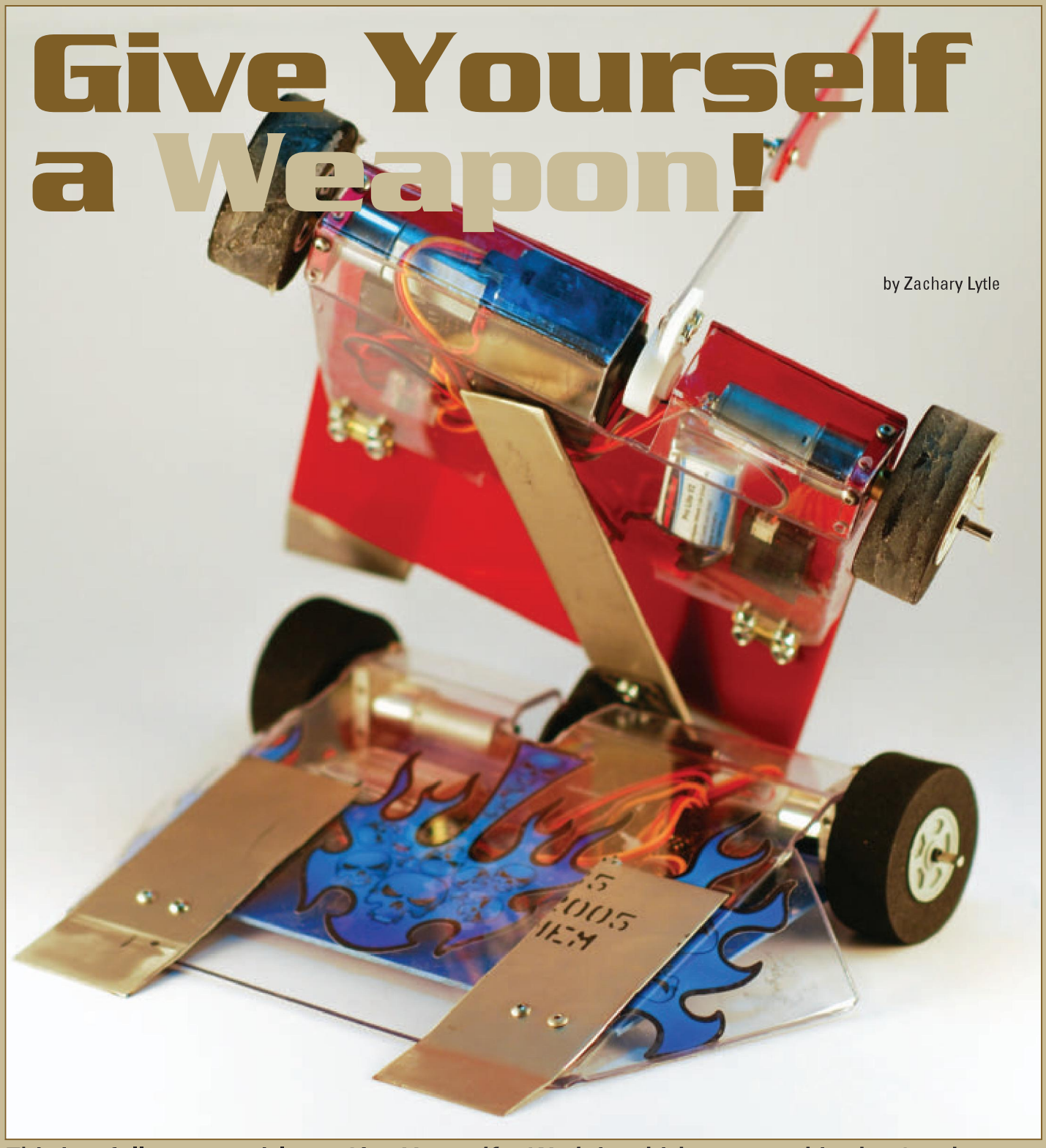
THE COUNTDOWN HAS BEGUN

Lynxmotion is a trademark of
RobotShop Distribution inc.

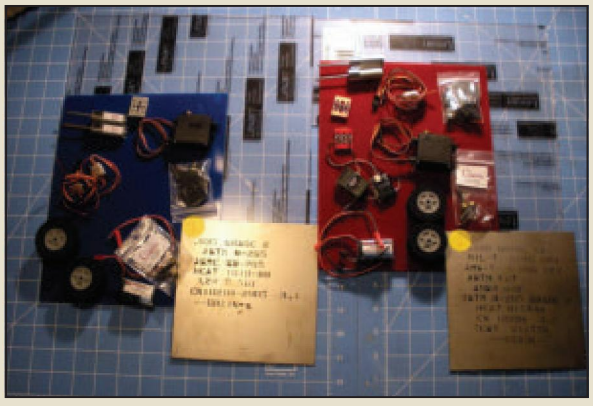
RobotShop.com
Robotics at Your Service!™

Give Yourself a Weapon!

by Zachary Lytle



This is a follow-up article to *Give Yourself a Wedgie* which appeared in the October 2012 issue of *SERVO Magazine*. In this article, I will be discussing how to add a servo weapon to your one pound fighting robot. As a four-time Insect champion at RoboGames, I have built and seen all kinds of servo driven weapons. I will be showing you two of the most effective ones: the lifter and the clamp. Both designs have the same internals and electronics. The only variation is in the assembly. So, I will start by showing you how to wire a servo into a one pound fighting robot's electronics. Then, the article will split into the "clamp build" and the "lifter build."



The materials you'll need are:

1. Two Fingertech ESC speed controllers
2. Two Fingertech 22-to-1 motors
3. Two Fingertech aluminum motor mounts
4. One Li-Po battery 7.4 volt, 320 milliamps
5. One metal gear servo capable of handling 7.4V
6. One Spectrum DX5-e radio
7. One Spectrum receiver
8. 6" x 6" titanium sheet, .032" thick
9. 8" x 11" polycarbonate sheet, .0925" thick
10. 8" x 11" Garolite sheet, .078" thick
11. One rigid L piece 1.5" x 2" (lifter only)
12. Servo saver mid-size (clamp only)
13. Two brass hinges
14. Four 2-56 screws; Allen head
15. 40 button head 4-40 screws
16. 10 4-40 locknuts
17. Heat shrink, 1/8" diameter
18. One pair Fingertech wheel hubs
19. One pair 2-1/4" Lite Flite tires
20. One female JST lead
21. Li-Po charger

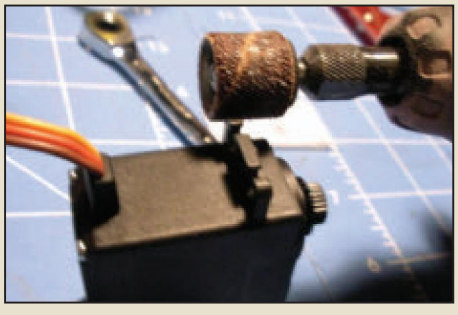
All of these materials are available on Fingertech website at **www.fingertechrobotics.com/products.php?cat=Robot+Kits**.

The tools you'll need are:

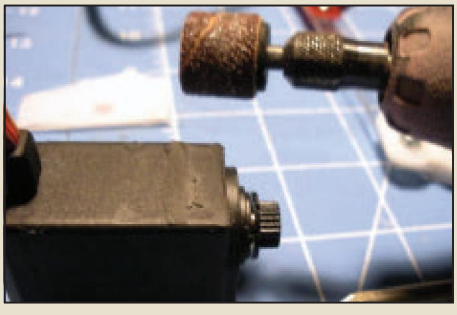
1. Soldering iron with solder
2. Hand drill with 1/8" bit
3. Pair of heavy duty shears
4. 3/32 and 1/16 Allen wrenches
5. Six inch caliper
6. Hammer
7. Punch or sturdy nail
8. Spray can of WD-40
9. Dremel tool with cutoff wheel and sanding wheel
10. Metal file
11. Wire cutters
12. CA glue

Optional *(but helpful)*:
Drill press
Belt sander
Heat gun

1. We need to modify our servo so that it can fit flush against the back plate.

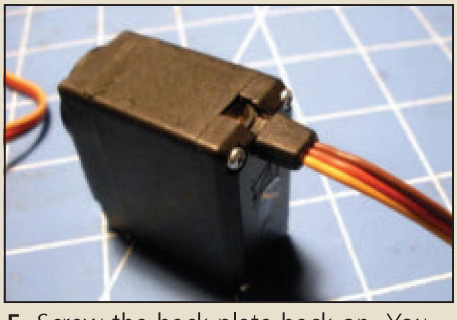


2. To do this, we'll start by sanding off the black clip closest to the servo horn.

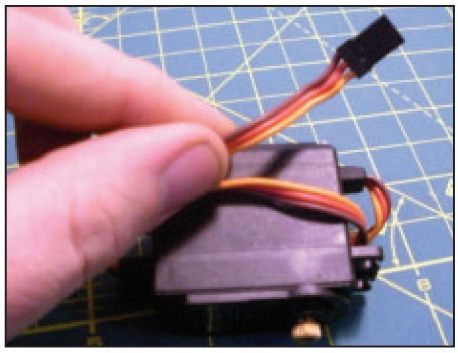


3. Use your Dremel tool with either a cutoff wheel or a sanding wheel to cut off the black clip.

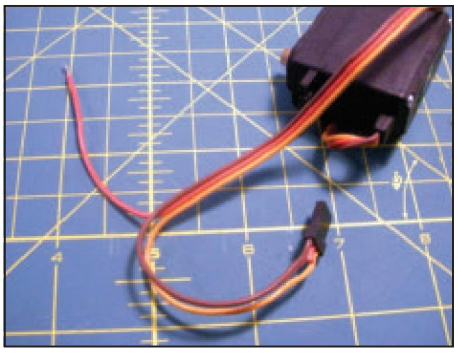
4. Next, we will cut a slot so the wires can run out the back of the servo and not the top. Remove the four screws on the back of the servo, then remove the back plate. Elongate the existing hole with your Dremel tool using the cutoff wheel.



5. Screw the back plate back on. You now have a modified servo.

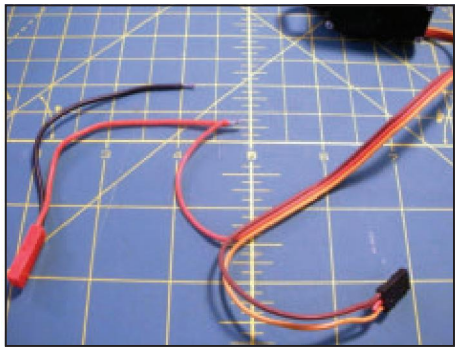


1. We will start out by wiring a metal geared servo — capable of handling 7.4 volts — to run directly off your robot's battery. It is important to run off the main battery and not the receiver power to avoid brownouts and shutdowns in your robot.

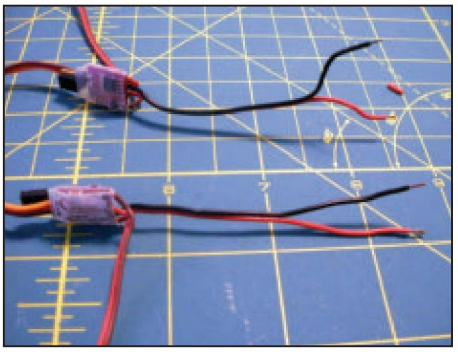


2. Start by cutting the red wire attached to the servo. This is tricky because it is the middle wire. You will leave the yellow and brown wires alone and attached to the receiver plug. It is important to not cut or disturb the brown wire because it is the ground. In order to have proper communication between the receiver and servo, you need the ground and signal wires connected.

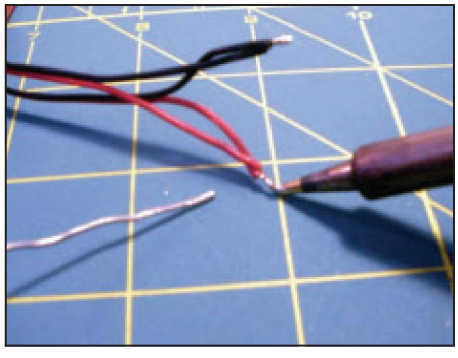
WIRING

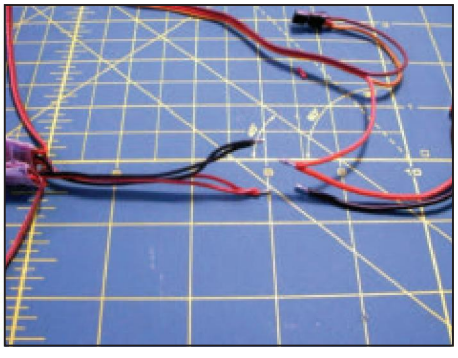


3. Next, join the red servo lead to a female JST lead.

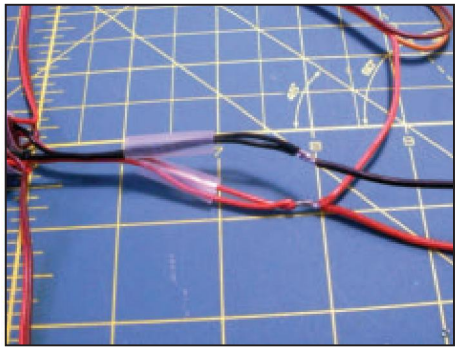


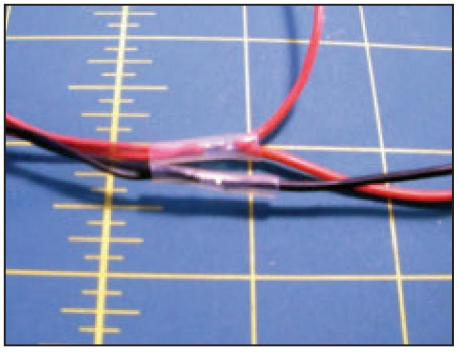
4. Prep two FingerTech speed controllers. Using a similar method to what we used in *Give Yourself a Wedgie*, strip the red and black leads, and tin them with your soldering iron.



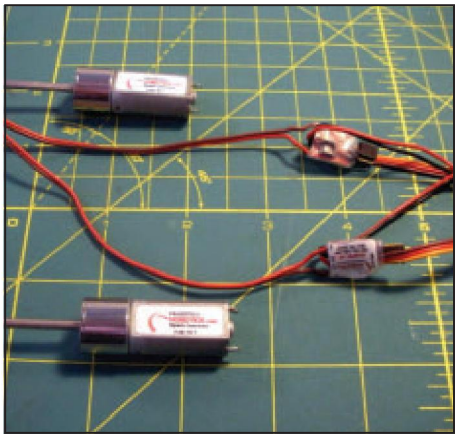


5. Now, join the two red leads on the controllers, then join the two black leads together. Slip a piece of heat shrink over the two red and two black leads.





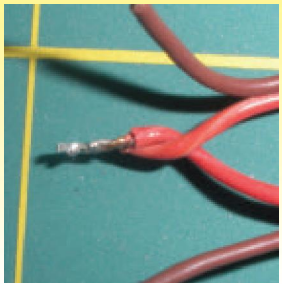
6. Solder all four wires together in one soldering joint, then slip the heat shrink over the joint and use a lighter to tighten it. Repeat the process for the two black wires from the controllers and the black wire on the JST plug.



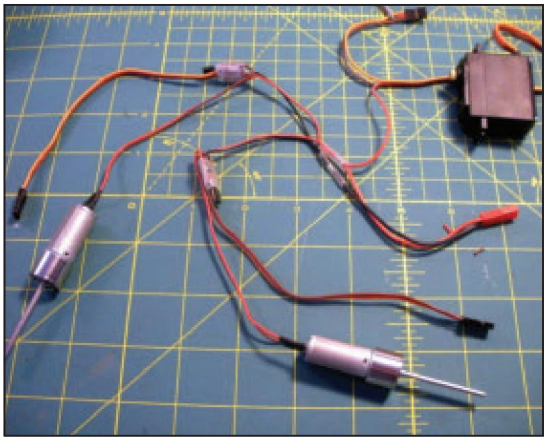
7. The black set of wires will only have three wires in the joint, as opposed to four in the red joint.

8. Next, tin and solder the red and brown leads from the speed controllers, then tin the two motor leads on the Silver Spark geared motors. Slip a piece of heat shrink over both the red and brown leads, then solder the red lead to the positive tab on the motor.

BUILDER TIP:
When soldering groups of wires together, it is best to twist two of them together before you tin them.

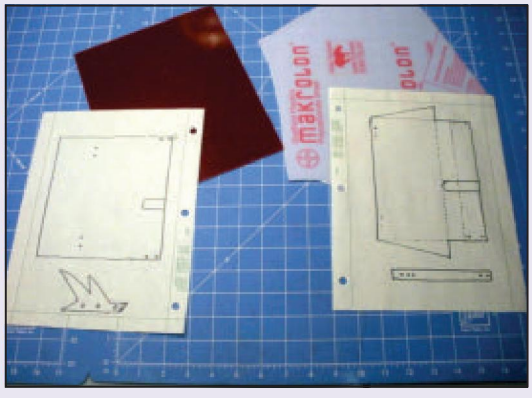


9. Solder the brown lead to the negative tab. Once you've soldered the wires to the motor leads, pull the heat shrink over the top of the solder joint and use the lighter to tighten it.

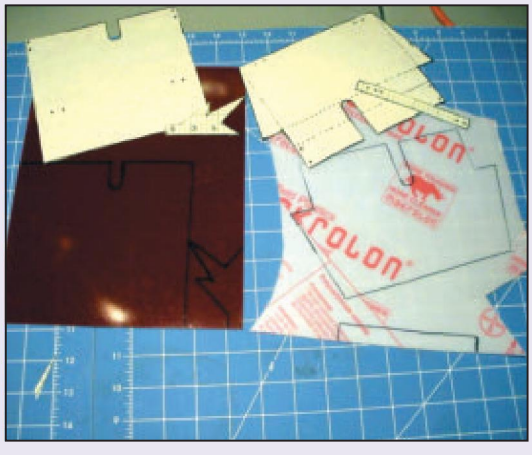


10. You have now completed the wiring for a weapon-based fighting robot. You must now make the all important decision: Will you be making a clamp or a lifter?

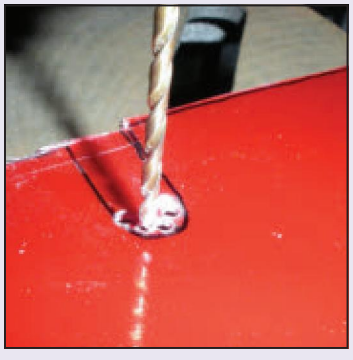
CLAMP



1. We'll start with the easier of the two builds, which is the clamp. When building a clamp robot, it is important to have a rigid frame in order to get a good "bite" on your opponent. For this purpose, I am starting with a sheet of Garolite.

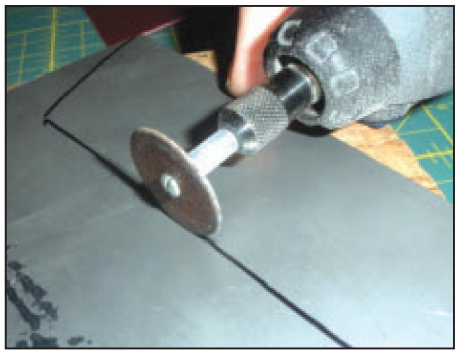


3. There are two pieces to make out of all three materials. Start with a Sharpie and trace the parts onto their respective sheets.

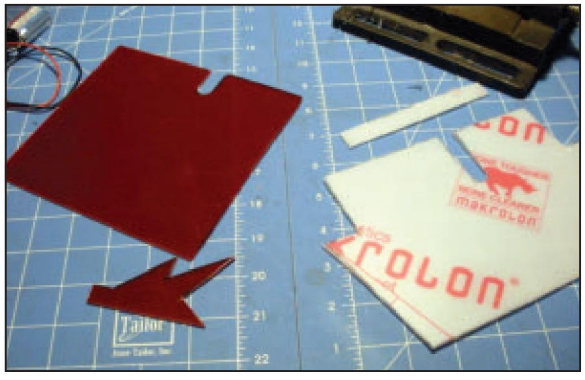


4. You will be able to cut the Garolite and the Lexan using shears. To prevent sharp corners which easily tear, drill holes in order to make a radius in the corners of your parts.

2. Garolite is similar to fiberglass in that it is rigid and lightweight. I will be using this as the main frame piece. I will also be using a polycarbonate sheet — better known as Lexan — for the underbelly. The armor plates will be 0.06 thick titanium. A template for the design is provided at the article link. Start by printing it out and cutting out the shapes.

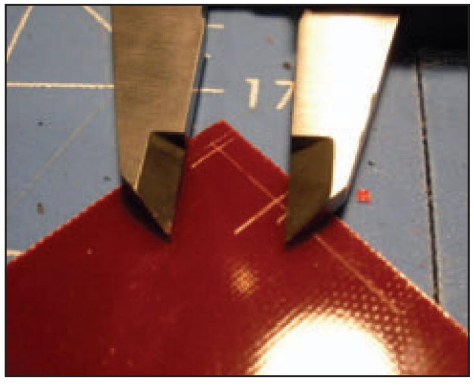


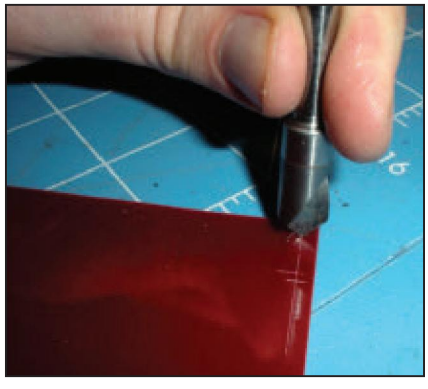
5. For the titanium, however, you will need a Dremel tool with a cutoff wheel. I recommend cutting on the outside of the black Sharpie line. Use your drum tool or a file to shave off the extra.

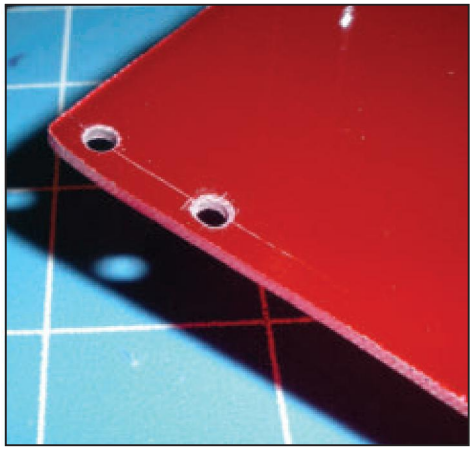


6. Once you have completed cutting out all six pieces, it is time to start the long drilling process. I recommend starting with the primary Garolite sheet.

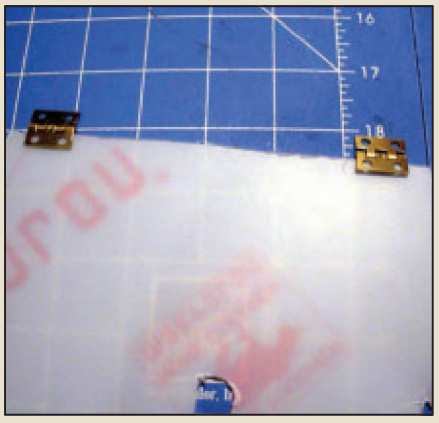
BUILDER TIP:
Keep the scrap pieces! If you need a last-minute weapon or need to patch a hole in your robot, spare material comes in extremely handy.

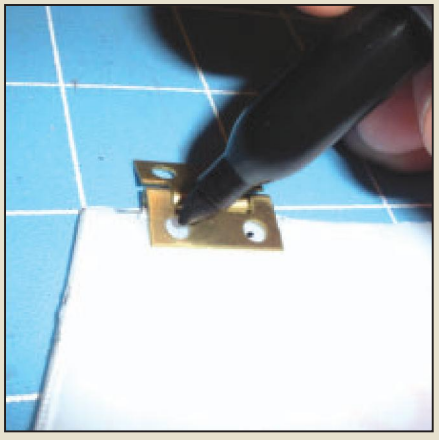


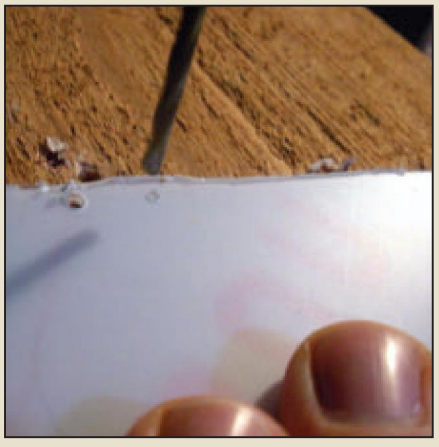




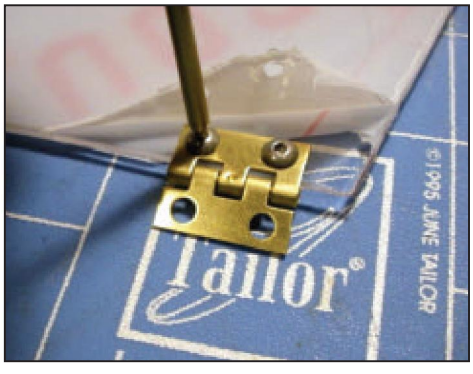
7. You will need to put eight holes in the sheet. Start with the two motor mounts on the edges; use the same caliper, scribe, and punch and drill method we used in the previous article.



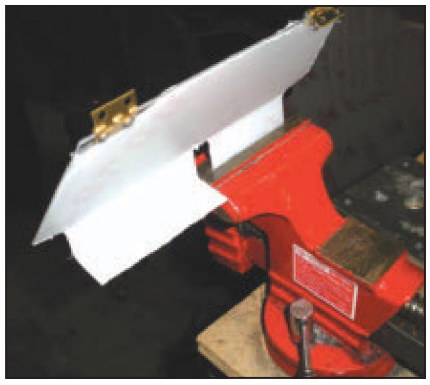




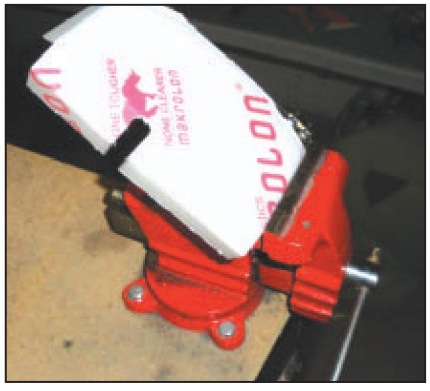
8. Now, drill four small holes along the leading edge for the mini hinges. Use the same scribe and punch and drill method discussed previously.

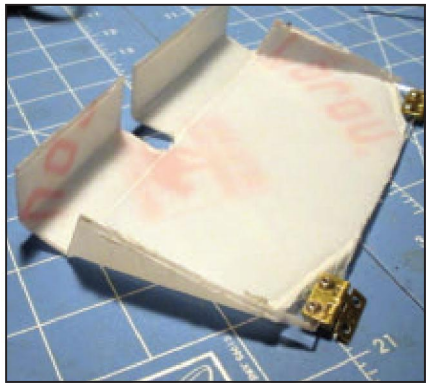


9. Use four 4-40 button head screws to attach the hinges to the Lexan plate.

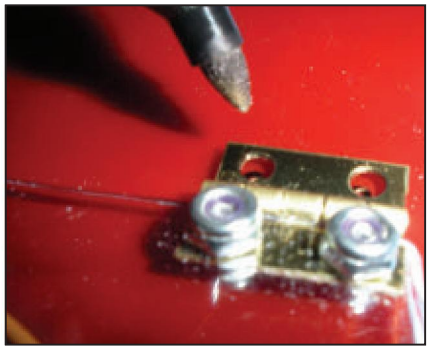


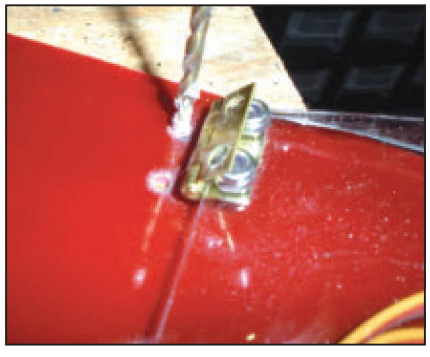
10. Next, bend the Lexan sheet along the dotted lines. Start with the 15 degree bend in the middle, then do the 90 degree bend along the back.



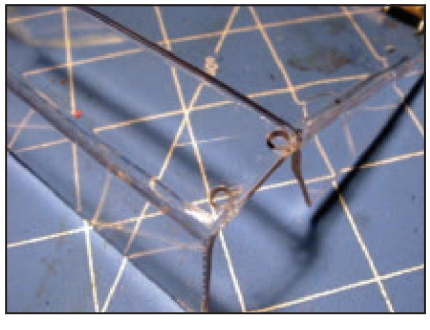


11. Finish with the two 90 degree bends along the right and left sides. After completing this, do a test fit to make sure the Lexan sheet fits over the motor mounts.





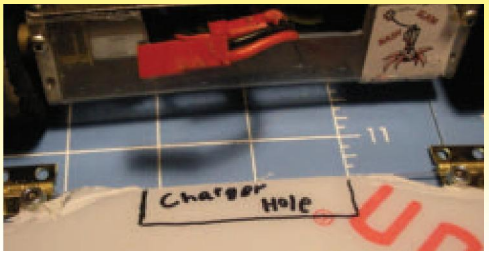
12. After test fitting (and possibly tweaking), we will now drill holes in the Garolite sheet for the mini hinges. Place the Lexan sheet on top of the robot and use a Sharpie to trace the four hole positions. Then, use the punch and drill method to drill the holes in the Garolite sheet.



13. Now, peel the backing off the Lexan to reveal the translucent sheet. You should be able to see the tapped holes through it. Mark four of them with a Sharpie and use the punch and drill method to drill them out. Next, we'll move on to the weapon.

BUILDER TIP:

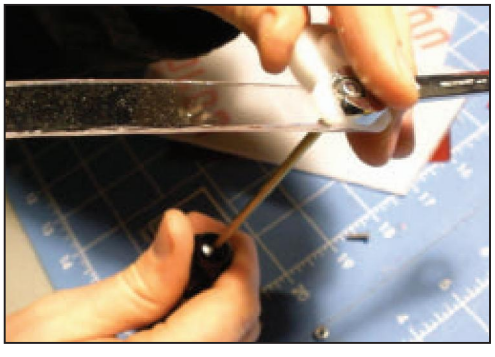
Usually at this point in the frame construction, I decide where the charging hole will be. You should try and hide it as much as possible.



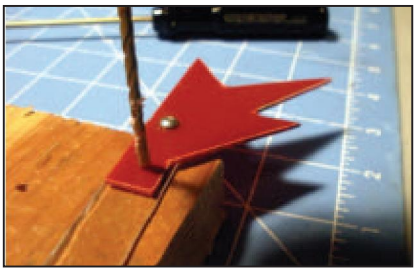
CLAMPING ARM

Now, let's build the clamping arm.

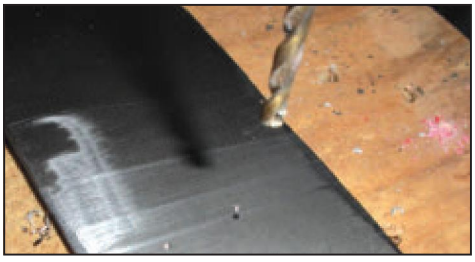
1. First, attach the Lexan arm to the servo saver. Start by drilling out two of the holes on the servo saver to 1/8". Place the servo saver on the end of the arm and trace the two holes with a black Sharpie onto the Lexan arm.



2. Drill out the arm with the punch and drill method. Use 4-40 screws and nuts to attach the two pieces together.



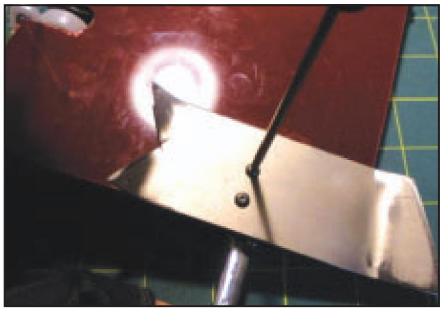
3.Next, attach the Garolite gripper to the other end of the Lexan arm. It should be noted that you can take artistic license with the gripper; you can make it out of titanium or Garolite. Drill two 1/8" holes that line up in both parts, then bolt them together using 4-40 screws and nuts.



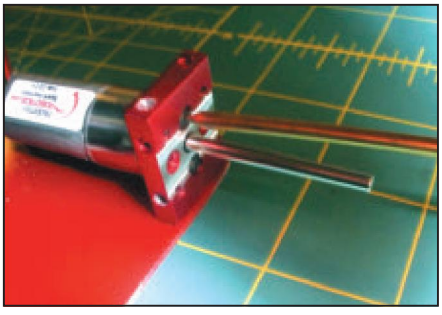
4. Next, drill two holes in the titanium plate that match up to the two holes in the hinges and Garolite. Although Garolite is rigid, it doesn't stand up to cutting saws. For this reason, I put titanium plates on the front corners of the wedge.

BUILDER TIP:

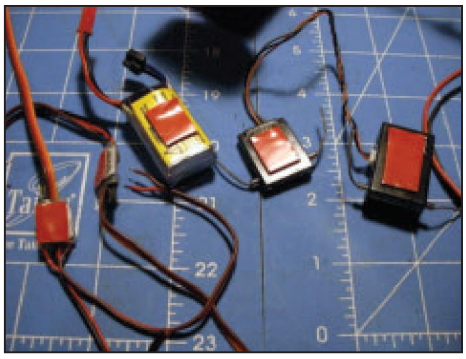
To speed up sharpening wedges or battle spikes, it is fastest to use a belt sander.

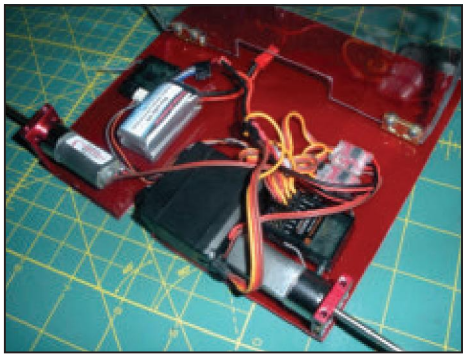


5. This is another step where you can take artistic license to help make this robot your own design. In my case, after drilling the holes, I bent the back corners to make titanium barbs and cut angles into the front wedge to give it a "V" shape. You have now completed the frame for a one pound clamp robot!

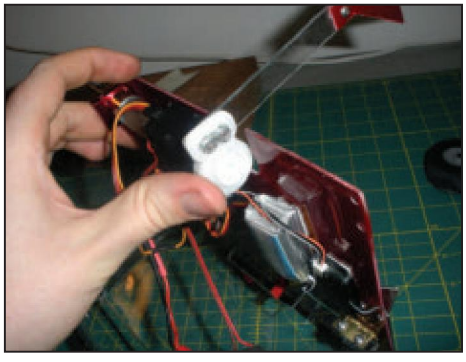


6. Now, we will add in the electronics that you soldered previously. Start by screwing the motors into the motor mounts. You will use two 2-56 screws in the front plate.

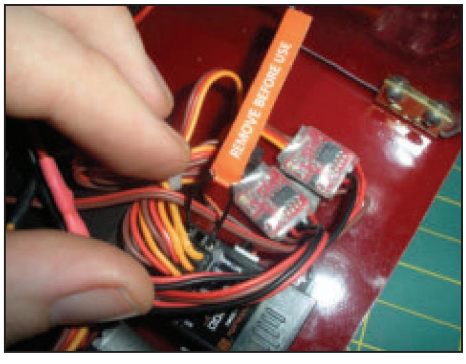




7. For your next step, you will tape down the two speed controllers, battery, and receiver in the positions shown.

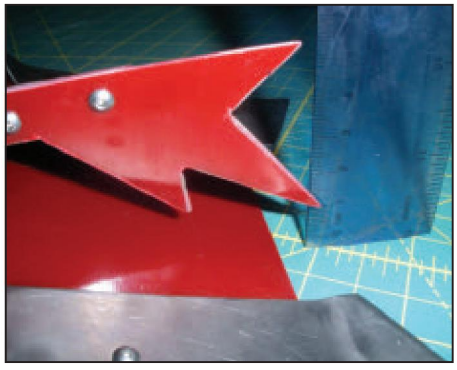


8. Next, slip the servo saver onto the servo. Before you tape the servo down, you will need to determine the travel of the servo.

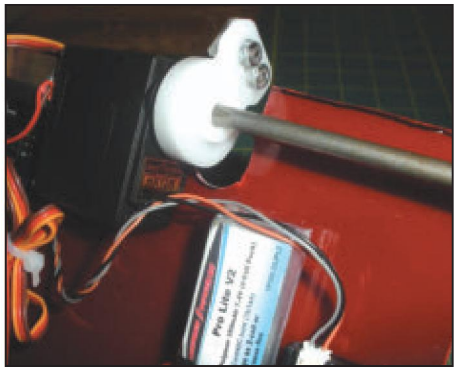


9. To do this, plug the servo lead into the throttle channel on your receiver. Then, plug the two speed controllers into the Elevator and Aileron slots on the receiver.

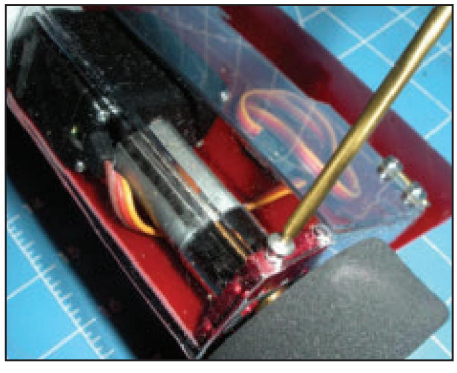
10. Bind the radio to the receiver and turn on the robot. The servo will immediately travel to where you have left the throttle stick on the radio.



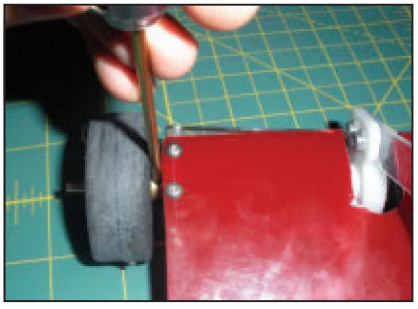
11. Move the throttle stick to the down position, then mount the arm on the servo so it's 1 cm from the bottom plate.



12. Once you are satisfied with the position of your clamping arm, screw the servo horn screw into the servo, locking the servo in the desired position.

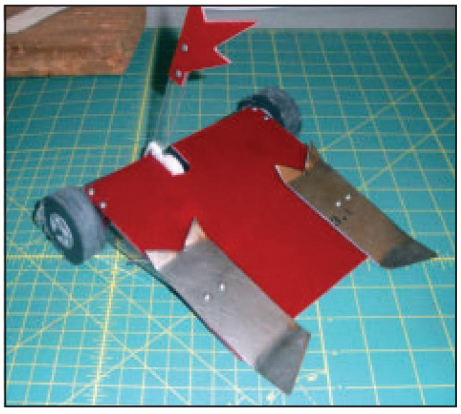


13. Screw down the bottom plate.



You have finished a one pound clamp robot!

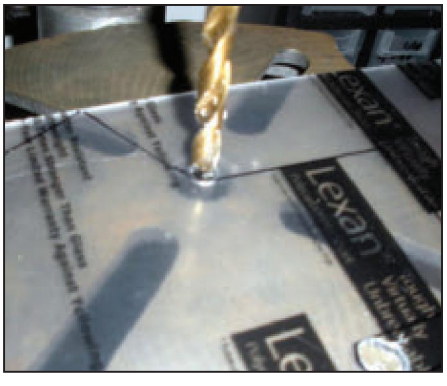
14. Now, mount the foam wheels using the wheel hubs. Put a small dab of glue on the hubs, then slide them into the foam wheels. Finally, use the set screw on the hub and screw it on to the driveshaft.



LIFTING

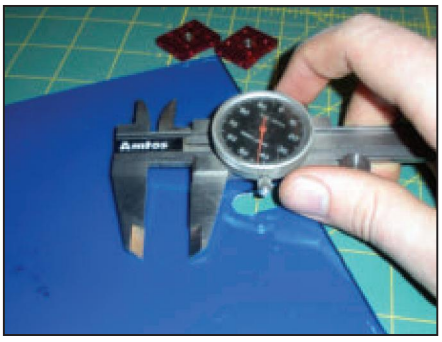
For the lifting robot, I invert the materials for a different approach. Although the clamp robot is rigid so it can get a tight grip on its opponent, I build lifters with a more flexible frame by using the Garolite as the rigid backbone and the Lexan as the outer skin. The Lexan can bend and absorb impacts which makes it excellent against axe, hammer, and spinning robots. For the lifter build, we will construct the undercarriage in Garolite and the top plate in Lexan. It will still need the titanium plates to prevent damage from cutting saws.

Print out the template (at the article link) for the lifting design. Cut the template out and trace the parts onto their respective sheets. This time, you should have three titanium plates, one Garolite base plate, and one large Lexan wedge. The last piece is a rigid L made out of either carbon fiber or steel.

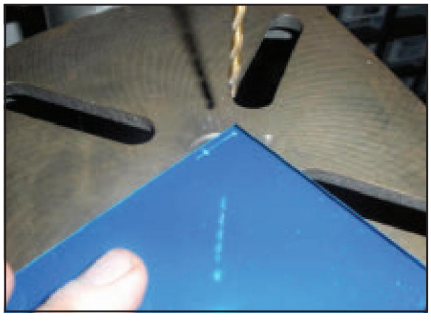


For the Garolite or Lexan, remember you can use shears to cut the template out. Use a 3/8" drill in the corners to create a radius.

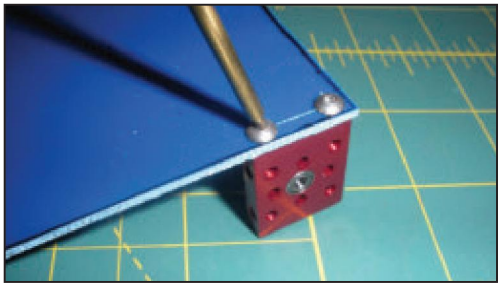
1. For the titanium, you will need to use a Dremel tool with a cutoff wheel. When cutting the parts, remember to cut on the outside of the Sharpie line. You can always file it down to the correct size.



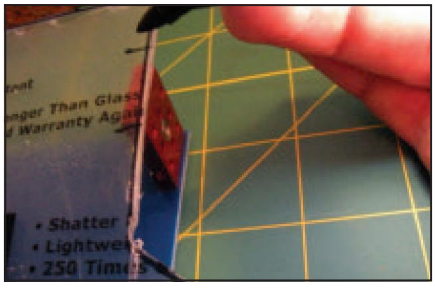
2. I would start next with the Garolite sheet and drill the holes for the two motor mounts. They will be on the top right- and left-hand corners. Use the scribe and punch and drill method.



3. After you have drilled the holes for the motor mounts, drill holes for the small hinges on the leading edge.



4. Next, use 4-40 screws and screw the motor mount and brass hinges to the base plate of your robot.

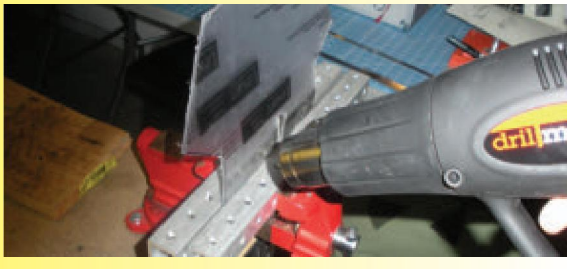


5. Take the Lexan sheet and lay it square over the frame piece.

6. Bend the Lexan along the dotted lines in a vice or between two blocks.

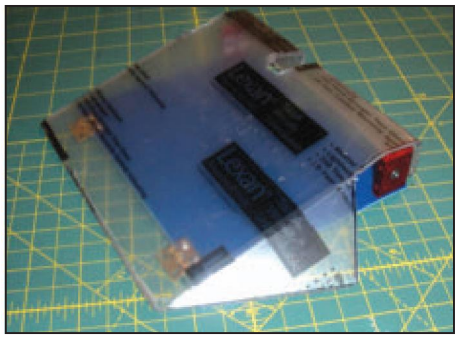
BUILDER TIP:

You can use a heat gun to help bend thicker plastics.

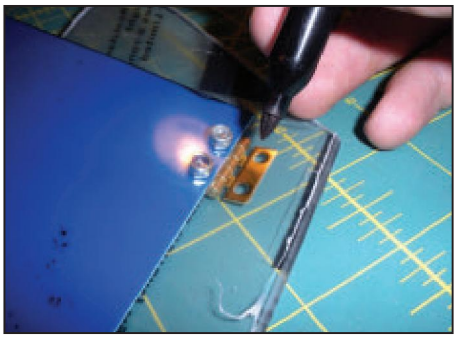


BUILDER TIP:

Know your titanium! There are two kinds you should use in fighting robots: Grade 2 and grade 5. You can bend grade 2 to create complex shapes, however, this means your robot can bend too. For indestructible armor, you should use grade 5 because it cannot be bent. You have to use it in its straight form, but it is some of the best armor you can put on your robot.



7. The sloped edges on the side of the wedge help deflect horizontal saws and shell spinners. By not giving your opponent an exposed edge, you make it harder to cut into the robot's wedge.

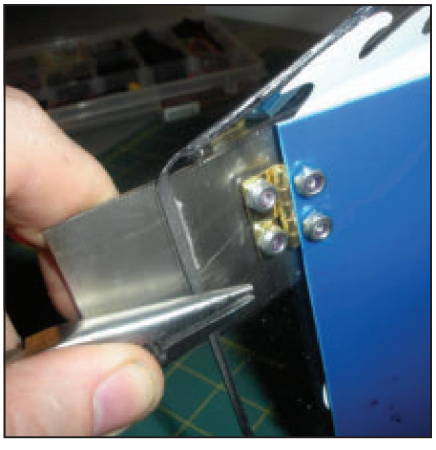


8. After the Lexan plate has been bent, mark the four holes for the hinges and drill the Lexan sheet with an 1/8 inch drill.

9. Before attaching the two sheets together, take the two shorter titanium pieces and line them up to where you would like them on your Lexan wedge.

10. Again, here's a step you can take artistic license with in order to give your robot its own personality. The holes you use for the titanium pieces are the same two holes used for the hinges. Trace the holes onto the titanium and use our usual punch and drill method.

11. A further step you can do to make this robot your own is adding internal stickers from your local hobby store. Internal stickers go on the *inside* of clear sheets so they won't be cut or ripped off by other robots. They will last as long as your armor holds out. (This step is optional, of course.)



12. After drilling out the titanium, reassemble the frame. Do this by screwing the Lexan, titanium, and hinge together with four 4-40 screws and nuts.

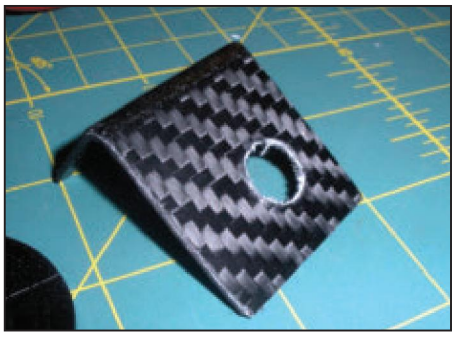
13. Lay the Lexan sheet over the two motor mounts. Since the Lexan is clear, you can see where the holes are and trace them.Once you have traced all four holes, remove the Lexan, and use the punch and drill method to drill out the four traced holes.

Now, we will work on the lifting arm.

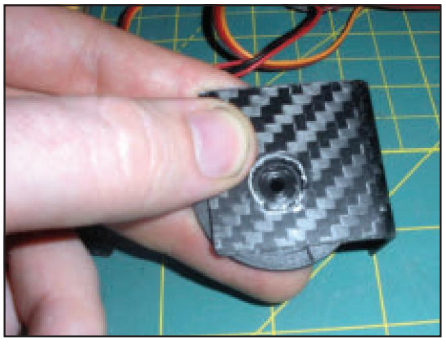
LIFTING ARM

1. Start by taking the steel or carbon fiber angle piece and drill a 0.425 hole on the left side.

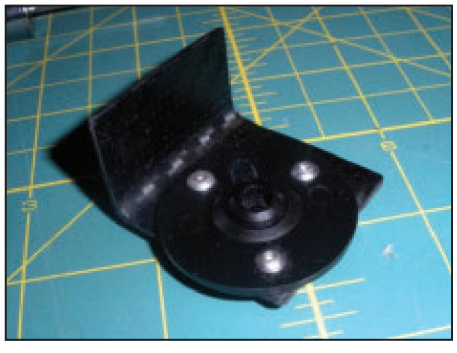
Before you drill the main clearance hole, take a moment to test-fit the servo arm and ankle piece. There are a large number of factors that could change the positioning of this hole. As opposed to altering the frame to make the template work perfectly, at this point, I would alter the placement of the hole to make sure the lifter fits.



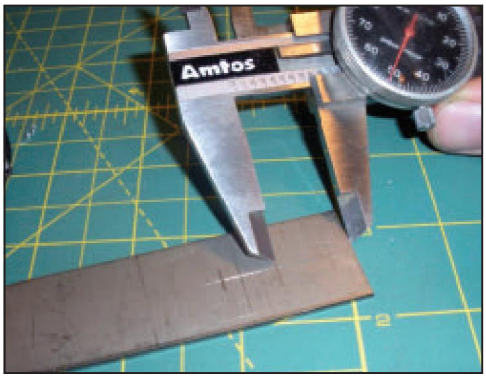
2. After locating and drilling the hole, place the servo horn that came with your servo in the .425 hole.



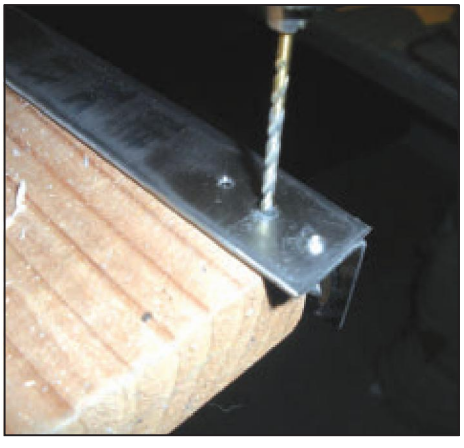
3. Pick three of the small holes on the servo horn and drill them out to 1/8". Drill three matching holes in the angle plate.



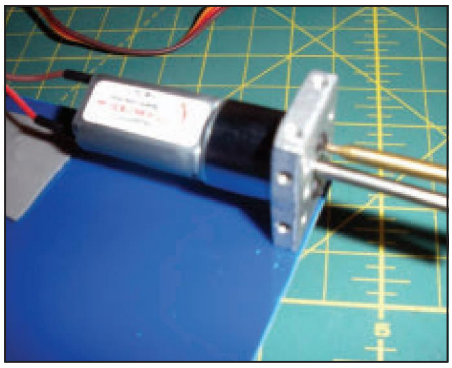
4. Put two 4-40 button head screws with the head on the angle plate side and the 4-40 nut on the servo horn side.



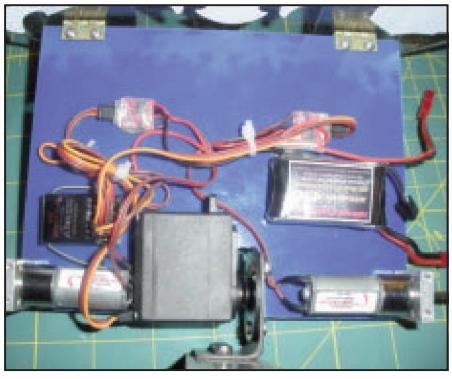
5. Once you have attached your servo horn, it is time to attach the titanium plate. Drill three holes in the titanium — 1/4", 3/4", and 1-1/2" from the top of the plate.



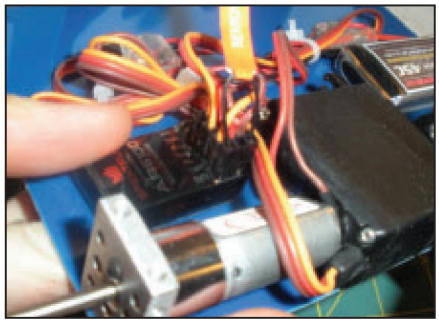
6. Put the titanium plate over the angle bracket and trace your three holes. Drill out all the holes with a 1/8" drill, and screw the angle bracket and the titanium plate together.



7. You are now ready to put the guts inside your robot! Start by screwing the motors into the motor mounts. You will use two 2-56 screws in the front plate.



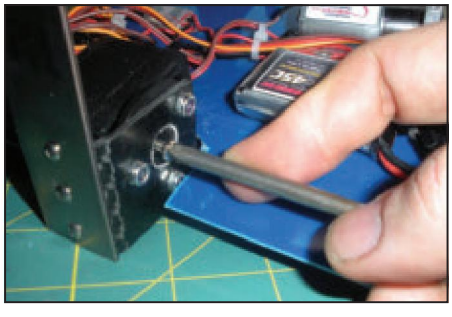
8. For your next step, tape down the two speed controllers, battery, and receiver in the positions shown in the **photo**.



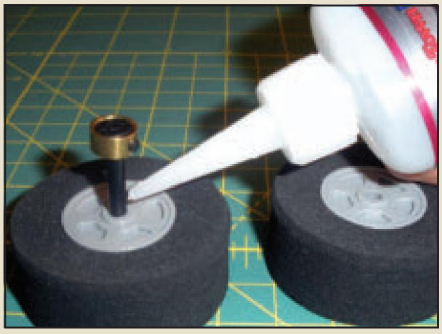
9. Next, slip the lifting arm onto the servo. Before you tape the servo down, you will need to determine the travel of the servo. To do this, plug the servo lead into the throttle channel on your receiver. Then, plug the two speed controllers into the Elevator and Aileron slots on the receiver.

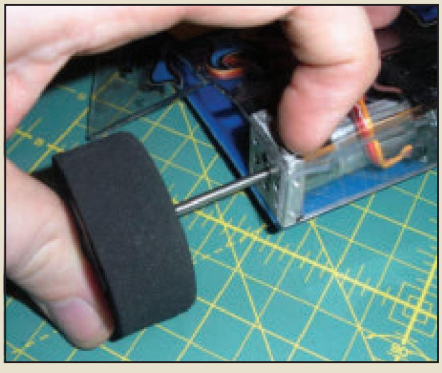
10. Bind the radio to the receiver and turn on the robot. The servo will immediately travel to where you have left the throttle stick on the radio.

11. Mount the lifting arm so that the leading edge is flush with your wedge.

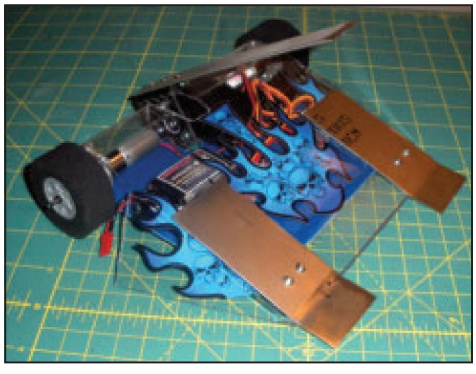


12. Then, put the servo horn screw in, locking the lifting arm in the desired position.



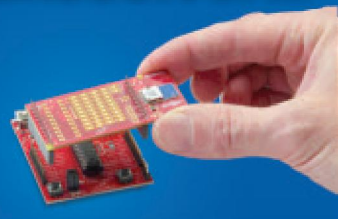


13. Screw down your top plate and mount the foam wheels using wheel hubs. Put a small dab of glue on the hubs, then slide them into the foam wheels. Finally, use the set screw on the hub and tape it on to the driveshaft.



You have now completed a one pound lifting robot!

A remarkably elegant low-power *transceiver...*



...and an equally elegant development tool to help you *use* it.

Introducing AIR-equipped BoosterPacks for the popular, new LaunchPad development tool from Texas Instruments!

- Each BoosterPack kit includes two AIR-equipped boards and two pre-flashed MSP430 MCUs. Just 'snap' them onto your LaunchPad and you'll be up and communicating in no time.
- Three AIR BoosterPacks are available today, with more on the way:

BOOSTER PACK	AIR MODULE	BAND/CERT
A1101R08A-LPF	A1101R08A	868MHz ETSI
A110R09A-LPF	A1101R09A	915MHz FCC/IC
430BOOST-CC110L*	A110LR09A	868/915MHz FCC/IC/ETSI

*Available only through TI e-store

- Prefer to use TI's EZ430 or DK development kits to work your app? We also offer AIR-equipped EZ target boards and EM boards to support those!

To learn more, write AIR@anaren.com, visit www.anaren.com/air, or scan the QR code with your smart phone.

Texas Instruments Design Network

Anaren®
What'll we think of next?™

800-411-6596
www.anaren.com
In Europe, call +44-2392-232392

CONCLUSION

After finishing your one pound weapon robot, it is important to practice. I recommend leaving your original Wedgie robot (from the October 2012 article) as-is if you still have it, then have both robots ready to fight so you can practice battling the two robots together with friends and family. This is a great way to get time behind the stick, and possibly get your friends into fighting robots.

A piece of advice that was given to me by a Battlebots veteran was, "In order to win, you need three things: a good build, a good driver, and some luck." Through a decade in the sport, I can say — without a doubt — this is 100% true. By going through two completely different robots, your second one will be much more polished. By having two robots, you can practice, so you will be a better driver. Lastly, some dedication and time will help you beat the odds.

With the skills you've learned in this article, you'll be able to build just about any kind of lifting or clamping robot you can imagine. I've included a few of my builds to show different ways to use these techniques in the **photo**.

I hope this gets you off to a smashing start in competitive robotics. I will see you in the ring! **SV**

www.servomagazine.com/index.php?/magazine/article/april2013_Lytle

Discuss this article in the SERVO Magazine forums at http://forum.servomagazine.com

ROBOGAMES

World's Largest Robot Competition
-Guinness Book of Records

North America's Top Ten Geek Fests
-Wired Magazine

SportCenter's Top Ten
-ESPN SportsCenter

"If you are a robot enthusiast, I would definitely encourage you to attend the RoboGames... Take a plane, train, space elevator, but definitely go!"
-Servo Magazine

"Impossible to Imagine, Impossible to Forget!"
-Robot Magazine

Compete at RoboGames!
Last year, over 1000 builders from around the world brought over 800 robots to San Francisco, in the 4th annual international event. This year, we expect even more robots and engineers to compete. Be one! With 80 different events, there's a competition for everyone - combat, androids, sumo, soccer, Lego, art, micromouse, BEAM, or Tetsujin! More than half the events are autonomous. Even if you just come to watch, you'll be overwhelmed with the diversity.

Last year, RoboGames hosted teams with over 800 robots from Argentina, Australia, Austria, Brazil, Canada, China, Colombia, Czech Republic, Denmark, Germany, India, Indonesia, Iran, Japan, Korea, Mexico, Netherlands, Peru, Singapore, Slovenia, Sweden, Taiwan, UK, and the USA.

Be a RoboGames Sponsor!
RoboGames is the world's largest open robot competition - letting people of any age, gender, nationality, or affiliation compete. Sponsoring RoboGames not only helps more people to compete, but also gets your company unrivaled press coverage and visibility. The event has been covered by CNN, ESPN, Fox, CBS, ABC, NBC (live), EBS Korea, NHK Japan, BBC, and countless print and web companies. Your logo can be everywhere the cameras turn!

Rent a Booth!
Booth spaces are at the front of the venue, ensuring lots of traffic. With 3000-5000 people each day, you're company will get amazing traffic!

Events:

Combat: 340 lbs, 220, 120, 60, 30, 3, & 1 lbs

Androids: Wrestling, Demonstration, Stair Climbing, The Eagle, Door Opening, The Toss, BasketBall, Lift and Carry, Marathon, Obstacle Run, Penalty Kick, Dash, 3:3 Soccer, Weight Lifting

Open Events: Fire-Fighting, Robomagellan, Maze/MicroMouse, Walker Challenge, Biped Race, Robot Triathlon, Line Slalom, Ribbon Climber, Vex Open, Lego Challenge, Lego Open, Aibo Performer, Balancer Race, Best of Show, Bot Hockey

Sumo: 3kg - Auto & R/C, 500g , 100g , 25g, Humanoid

Robot Soccer: Biped 3:3 & 5:5 , Mirosot 5:5 &11:11

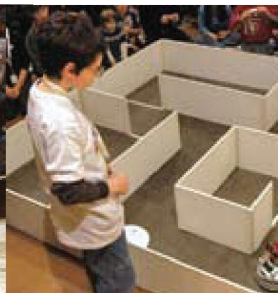
Junior League: Lego Challenge, Lego Open, Lego Magellan, Woots & Snarks, Handy Board Ball, BotsketBall, 500 g Sumo, 120 lb combat, Best of Show, Vex Open

Tetsujin (ExoSkeleton): Lifting, Walking, Carrying

Art Bots: Static, Kinetic, Bartending, Musical, Drawing

BEAM: Speeder, Photovore, RoboSapien Hacker

April 19-21,2013
San Mateo, CA
http://RoboGames.Net



The SERVO Webstore

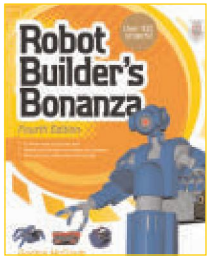
Attention Subscribers ask about your discount on prices marked with an *

CD-ROM SPECIALS

04.2013 FEATURED

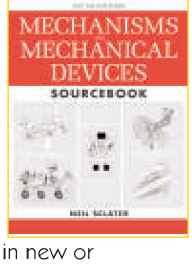
ROBOTICS

Robot Builder's Bonanza, Fourth Edition
by Gordon McComb



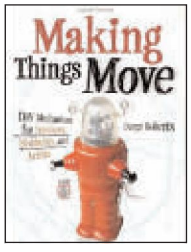
Robot Builder's Bonanza, Fourth Edition includes step-by-step plans for a number of motorized platforms. The book is described as a compendium of robotics topics, containing more than 100 projects, including 10 robot designs new to the fourth edition. These modular robots are low cost, and are made to be reproduced by readers with no training in mechanical construction.
$29.95*

Mechanisms and Mechanical Devices Sourcebook 5th Edition
by Neil Sclater



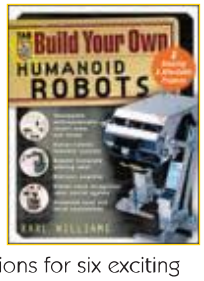
Fully revised throughout, this abundantly illustrated reference describes proven mechanisms and mechanical devices. Each illustration represents a design concept that can easily be recycled for use in new or modified mechanical, electromechanical, or mechatronic products. Tutorials on the basics of mechanisms and motion control systems introduce you to those subjects or act as a refresher.
Reg $89.95 Sale Price $79.95

Making Things Move: DIY Mechanisms for Inventors, Hobbyists, and Artists
by Dustyn Roberts



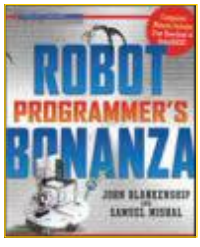
In *Making Things Move: DIY Mechanisms for Inventors, Hobbyists, and Artists*, you'll learn how to successfully build moving mechanisms through non-technical explanations, examples, and do-it-yourself projects — from kinetic art installations to creative toys to energy-harvesting devices. Photographs, illustrations, screenshots, and images of 3D models are included for each project.
$29.95*

Build Your Own Humanoid Robots
by Karl Williams
GREAT 'DROIDS, INDEED!



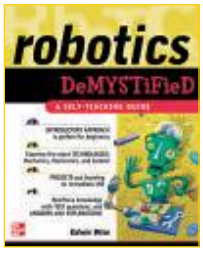
This unique guide to sophisticated robotics projects brings humanoid robot construction home to the hobbyist. Written by a well-known figure in the robotics community, *Build Your Own Humanoid Robots* provides step-by-step directions for six exciting projects, each costing less than $300. Together, they form the essential ingredients for making your own humanoid robot.
$24.95*

Robot Programmer's Bonanza
by John Blankenship, Samuel Mishal



The first hands-on programming guide for today's robot hobbyist!
Get ready to reach into your programming toolbox and control a robot like never before! *Robot Programmer's Bonanza* is the one-stop guide for everyone from robot novices to advanced hobbyists who are ready to go beyond just building robots and start programming them to perform useful tasks.
$29.95

Robotics Demystified
by Edwin Wise



YOU DON'T NEED ARTIFICIAL INTELLIGENCE TO LEARN ROBOTICS!
Now anyone with an interest in robotics can gain a deeper understanding — without formal training, unlimited time, or a genius IQ. In *Robotics Demystified*, expert robot builder and author Edwin Wise provides an effective and totally painless way to learn about the technologies used to build robots! **$19.95**

We accept VISA, MC, AMEX, and DISCOVER
Prices do not include shipping and may be subject to change.

To order call 1-800-783-4624

SERVO Magazine Bundles

Save $10.00 Only $57.95!

Now you can get one year's worth of all your favorite articles from *SERVO Magazine* in a convenient bundle of print copies. Available for years 04, 05, 06, 07, 08, 09, 10, 11 and 2012.

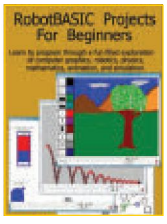
Any bot builders out there? Get cool robotics stuff from my store!

Call me at my order desk!

Visit my online store @ www.servomagazine.com

RobotBASIC Projects For Beginners

by John Blankenship, Samuel Mishal

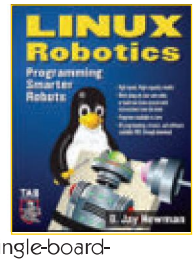


If you want to learn how to program, this is the book for you. Most texts on programming offer dry, boring examples that are difficult to follow. In this book, a wide variety of interesting and relevant subjects are explored using a problem-solving methodology that develops logical thinking skills while making learning fun. RobotBASIC is an easy-to-use computer language available for any Windows-based PC and is used throughout the text.

Price $14.95

Linux Robotics

by D. Jay Newman

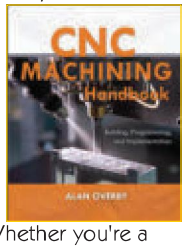


If you want your robot to have more brains than microcontrollers can deliver — if you want a truly intelligent, high-capability robot — everything you need is right here. *Linux Robotics* gives you step-by-step directions for "Zeppo," a super-smart, single-board-powered robot that can be built by any hobbyist. You also get complete instructions for incorporating Linux single boards into your own unique robotic designs. No programming experience is required. This book includes access to all the downloadable programs you need.

$38.95

CNC Machining Handbook: Building, Programming, and Implementation

by Alan Overby

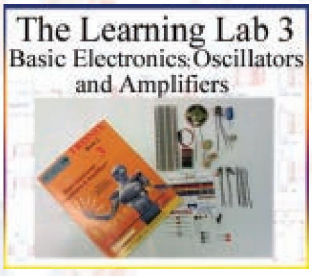


The *CNC Machining Handbook* describes the steps involved in building a CNC machine and successfully implementing it in a real world application. Helpful photos and illustrations are featured throughout. Whether you're a student, hobbyist, or business owner looking to move from a manual manufacturing process to the accuracy and repeatability of what CNC has to offer, you'll benefit from the in-depth information in this comprehensive resource. **$34.95**

FOR BEGINNER BOT BUILDERS

$59.95

$49.95



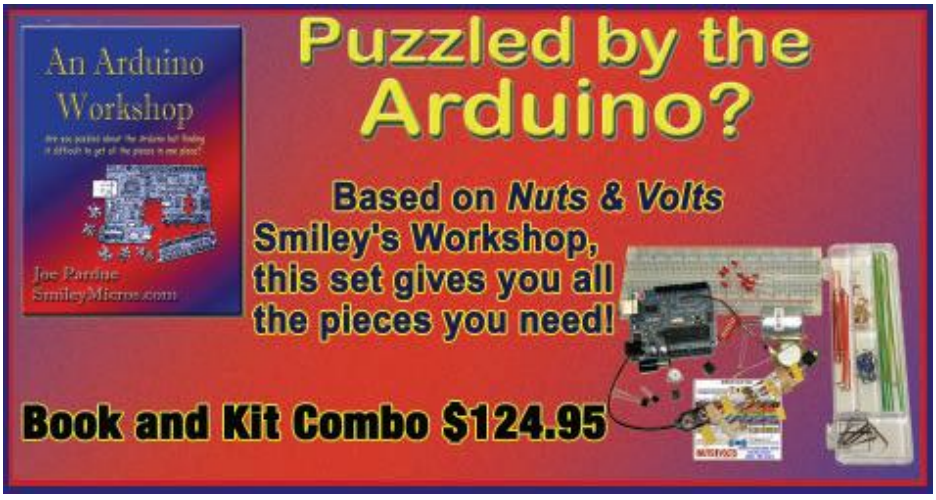
$39.95

The labs in this series — from GSS Tech Ed — show simple and interesting experiments and lessons, all done on a solderless circuit board. As you do each experiment, you learn how basic components work in a circuit, and continue to build your arsenal of knowledge with each successive experiment.

For more info and a promotional video, please visit our webstore.

Or order online
www.servomagazine.com

SPECIAL OFFERS



Forbidden LEGO

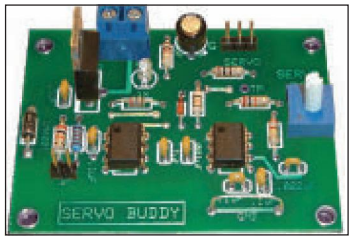
by Ulrik Pilegaard / Mike Dooley

Forbidden LEGO introduces you to the type of free-style building that LEGO's master builders do for fun in the back room. Using LEGO bricks in combination with common household materials (from rubber bands and glue to plastic spoons and ping-pong balls) along with some very unorthodox building techniques, you'll learn to create working models that LEGO would never endorse.

Reg $24.95 Sale Price $19.95

PROJECTS

The SERVO Buddy Kit



An inexpensive circuit you can build to control a servo without a microcontroller.

For more information, please check out the May 2008 issue or go to the SERVO webstore.

Includes an article reprint.
Subscriber's Price **$39.55**
Non-Subscriber's Price **$43.95**

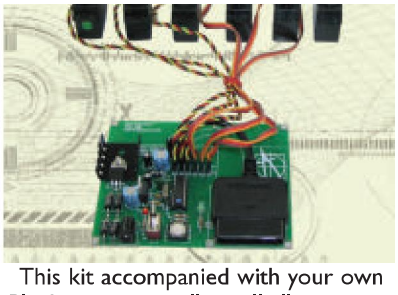
3D LED Cube Kit

From the article "Build the 3D LED Matrix Cube" as seen in the August 2011 issue of *Nuts & Volts Magazine*.

This kit shows you how to build a really cool 3D cube with a 4 x 4 x 4 monochromatic LED matrix which has a total of 64 LEDs. The preprogrammed microcontroller that includes 29 patterns that will automatically play with a runtime of approximately 6-1/2 minutes. Colors available: Green, Red, Yellow & Blue. Jig and plastic cases also available.

Subscriber's Price **$57.95**
Non-Subscriber's Price **$59.95**

PS2 Servomotor Controller Kit



This kit accompanied with your own PlayStation controller will allow you to control up to six servomotors. Includes all components and instruction manual.

For more information, please see the February 2011 edition of *SERVO Magazine*. Assembled units available!

Subscriber's Price
$79.95
Non-Subscriber's Price
$84.95

NASA Engineers Building a Lunar Mining Robot

by Kevin Berry

www.servomagazine.com/index.php?/magazine/article/april2013_Berry
Discuss this article in the SERVO Magazine forums at http://forum.servomagazine.com

After decades of designing and operating robots full of scientific gear to study other worlds, NASA is working on a prototype that leaves the delicate instruments at home in exchange for a sturdy pair of diggers, plus the reliability and strength to work all day, every day for years.

Dubbed RASSOR — for Regolith Advanced Surface Systems Operations Robot, and pronounced "razor" — the autonomous machine is far from space-ready, but the earliest design has shown engineers the broad strokes of what their lunar soil excavator needs in order to operate reliably.

The primary challenge for any digging robot operating off Earth is that they have to be light and small enough to fly on a rocket, but heavy enough to operate in gravity lower than that of Earth. RASSOR tackles this problem by using digging bucket drums at each end of its robot body that rotate in opposite directions, giving enough traction on one end to let the opposite side dig into the soil.

The team built a weight off-loading harness that simulated working the rover in the Moon's 1/6th gravity field. "We proved that if you engage one bucket, it pulls itself, but when you lower the other bucket and rotate it, once they both catch in, it starts digging," commented A.J. Nick, an engineer on the RASSOR team.

Another secret of the drum — inspired by a previous Lockheed Martin design — is the staggered shallow scoops that shave the soil a bit at a time, rather than scoop large chunks of it all at once the way bulldozers do on Earth.

During NASA's Lunar Regolith Challenges held every

FIGURE 1. The RASSOR robot climbs a hill during recent testing at NASA's Kennedy Space Center in Florida.

FIGURE 2. With a pair of drums positioned on the arms, the RASSOR can take on a number of different shapes to accomplish its work.

year at the Kennedy Space Center (as reported in *SERVO*), Embry Riddle's Prescott team demonstrated the "bucket drum" technique successfully in this student competition.

The RASSOR looks like a small tank chassis with a drum at either end, each attached with arms. The drums are perhaps the robot's most innovative feature. Because they are mounted on moving arms, they can act almost as legs letting the robot step and climb over obstacles. The team calls such moves "acrobatics." They point out that the robot can safely drive off the lander and right itself, flip itself over to get unstuck from fine soil, and lift the whole body off the ground to let its treads run smoothly to remove built-up soil. RASSOR is designed to easily make itself into a Z-shaped position to drop its soil collection into a hopper.

The robot is designed to skim lunar soil and dump it into a device that would pull water and ice out of the dirt and turn their chemicals into rocket fuel or breathing air for astronauts working on the surface. The device would be part of the lander that carries the RASSOR to the Moon's surface. So, the robot would be the feeder for a lunar resource processing plant — a level of industry never before tried anywhere besides Earth.

A concept mission for RASSOR would have a 2,000 pound payload in addition to the lander, which would be about the size of the Phoenix lander NASA sent to Mars. The RASSOR is expected to weigh about 100 pounds. The remaining payload would be used to process the lunar soil delivered by RASSOR.

Producing water and fuel from the lunar soil would save the tremendous expense of launching the supplies from Earth, since 90 percent of a rocket's mass normally consists of propellant — which could be made on the Moon.

In order to provide enough material to the production platform to create usable amounts of resources, the RASSOR would need to operate about 16 hours a day for five years. It would drive five times faster than the Mars Curiosity rover's top speed of four centimeters per second, then shave the Moon's surface with a pair of rotating drums, and return to

Information provided by NASA's Steven Siceloff, Kennedy Space Center. Photos provided by NASA.

the resource processing plant with some 40 pounds of lunar soil.

Devising a robot for such demands called for numerous innovations, and the team says it has at least one major decision to make before it begins construction of the second generation RASSOR prototype: keep going with tracks like those that tanks use or switch to wheels.

The tracks showed some flaws in recent testing, mostly relating to pebbles and sand particles clogging the gears and making the track slip off. The group tried out RASSOR on several surfaces at Kennedy, including the crushed river rock dug up from the crawlerway. The rock — even though pulverized by the gigantic crawlers — is not a great substitute for lunar soil, the engineers explained. As long as the robot handles that matter well, however, they say they know it will manage whatever the Moon soil offers.

Part of the problem (also mentioned) might be the rubber material the tracks are made of. A lunar version of the robot would use a different material, possibly metallic. For example, the lunar rover the astronauts drove on the surface used wheels made of stainless steel springs rather than rubber. (*SERVO* examined this topic in the article, *Dust In The Wheels* in the June 2011 issue).

The team already is designing RASSOR 2 — a prototype that would be much closer to what NASA could launch in the future. It's expected to begin testing in early 2014. **SV**

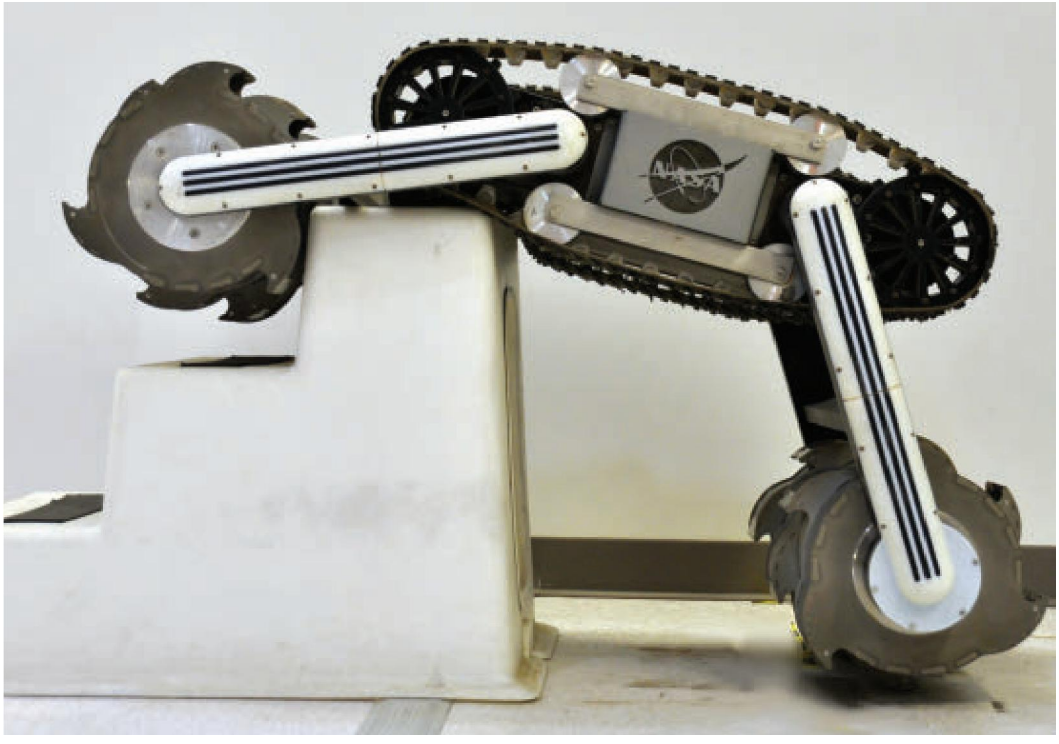


FIGURE 3. The RASSOR can climb over a large obstacle, such as a boulder on the moon. Engineers on Earth used a stepping stool to challenge the robot.

FIGURE 4. In the Lunar Regolith Challenge, Embry Riddle's Prescott team adapted a novel "bucket drum" technique originally concepted by NASA. *Photo courtesy of Joe Jacoby.*

Then and NOW

Get That Robot Built the Easy Way

by Tom Carroll

Robot building can be quite easy or very frustrating, depending on the approach you take in design and construction. Ten years ago, I wrote an article for the *Nuts & Volts* 'robot supplement' entitled *Robot Parts all Around Us*, to show potential builders just how easy it was to find robot parts. Parts to build great robots are all around us. There is a mind set that we must develop to be able to recognize these parts when we see them. When you are out shopping the next time, start to look around with a potential robot project in mind. Look past the item's intended use and imagine how it could be incorporated in a future robot project. "*Can this electric drill or this bowl or even this sliding door pulley be used when I build Robbie?*"

When personal and hobbyist robots first became popular 30 years ago and a few stores began to display some robot kits on their shelves, PCs were already well established. There were far more computers in homes than robots at that time. Robot enthusiasts realized that the earliest computers offered a way to write and then edit text, draw, and even develop complex spreadsheets. Hundreds of software programs became available for various needs.

However, robots at that time — despite their onboard computer — could do little more than roam about our living rooms, entertaining us with their cuteness.

Surplus houses in the '80s had vast stocks of gearmotors, metal stock, gears, drive belts and pulleys, unique hardware, and other items of interest to robot builders. It was the lack of functional — yet inexpensive — sensors coupled with cheap microcontrollers that held back the development of experimental robots.

These surplus outlets have changed since that time, and "big box" stores and Internet robot suppliers are now the sources for most of our home-built robots. PCs still outsell robots probably 25 to 1, as there are tens of thousands of stores that sell computers.

We don't have to worry any more since we have so many good sources for robot parts and even complete robots. However, before delving into how to locate parts for your robot, allow me to present two unique and first-rate home-built robots.

Dave Shinsel and Loki

I have mentioned these robots that epitomize extreme home-built robots in past articles. I first met Dave back in 2009 at a Portland Robotics meeting where he was demonstrating his homemade robot, Loki (shown in **Figure 1**). The body and arms are

FIGURE 1. Dave Shinsel and Loki.

made completely from scratch with items such as 99-cent flashlights; he used soup cans for the eyes.

Dave told me, "I am always on the lookout for useful interesting things that I can use in my robots. The shoulders of Loki are made from carpet protectors that go under a couch's legs. Loki's drive wheels are lawnmower wheels I picked up at the hardware store. His tail-wheel is made from two inline skate wheels I caught on sale from a sporting good store. Trim around the robot is mostly door-edge molding from an auto parts store, as well as the decorative lighting on the sides of the robot. There are lots of interesting decorative lights at auto parts stores."

"I also scrounge the local surplus stores. Loki's elbow rotation is supported by high quality bearing joints I picked up for $1 each. The shoulder motors use an expensive Kerr motor controller which I got surplus for $5, then learned how to program the thing from the company's website. Loki's blue shoulder brackets are also from the surplus store. Loki's left hand was a gripper for picking up things I got from a drug store. His voice comes from an iPod dock speaker mounted just below the head. The grills on the side of the robot are florescent light cover panels from Home Depot, painted black. The grill on the font of the robot's stomach is made from a bathroom vent cover (also painted black)."

Dave uses a laptop computer as the brains for Loki, and also as a chest-mounted display. The PC uses Visual C++ and an onboard PIC relies on the CCS C compiler. Two Logitech laptop cameras serve as eyes for his stereo vision system.

Typical of most robot builders, projects never end as improvements are always envisioned and added. **Figure 2** shows Loki's newer drive assembly using reliable Pittman gearmotors and a toothed belt to drive the wheels instead of the previous stepper motors.

There was a bit of machining required on the two 1/4" aluminum plates, but Dave used a heavy aluminum L extrusion to mount the plates. Iron angles would have interfered with the robot's compass.

That is the advantage of building your own robot from scratch — you can always change the structure and components as you locate better parts.

Dave has dramatically changed Loki from its birth in 2006 until now. Dave is a prime example of a robot builder who took common objects and made them fit into one of the finest robots that I have ever seen. Johnny Five from the film, Short Circuit was personable; however, I do believe Loki has far more actual personality than the movie action prop.

Dave and his daughter are 'Team Crash,' competing in the new SyFy series, Robot Combat League. For more information on Dave's many robotic creations, go to **www.dshinsel.com**.

Kenneth Maxon's NeRP

I first met Ken at one of the Seattle Robotics Society's Robothons. He had a most amazing creation which is shown in **Figure 3**. His large bot, Max, had 12 stepper motors for motion and over 4,600 screws to hold it together. **Figure 4** shows another remarkable robot that he has built.

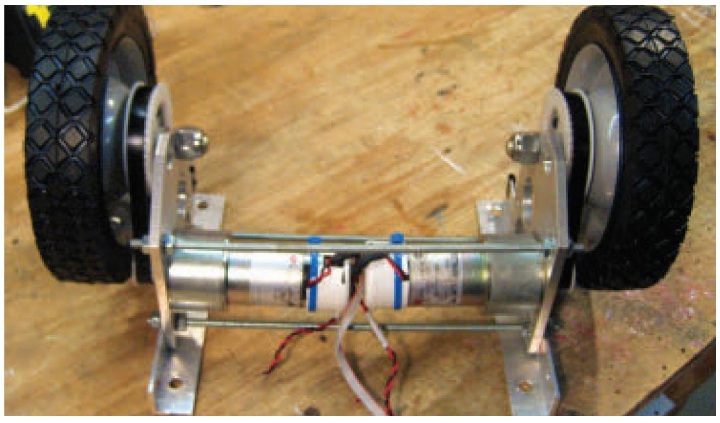


FIGURE 2. New wheels and motors for Loki.

Ken had a functional robot in mind when he designed it; one that could navigate around his block. Moreover, precision machining of the 6061-T6 aluminum stock into the shiny parts (shown in the **figure**) was critical to his design and to his desire for a truly beautiful work of art.

Figure 5 is a close-up of the robot's upper suspension. I could just stare at the robot for hours. **Figure 6** shows four views of the robot's ability to traverse obstructions using a rocker-bogey six-wheel system, similar to those used on the Mars rovers.

FIGURE 3. Kenneth Maxon's Max at Robothon.

Figure 7 is a partial view of Ken's shop showing a metal shear, large drill press, CNC milling machine, a large lathe, and other

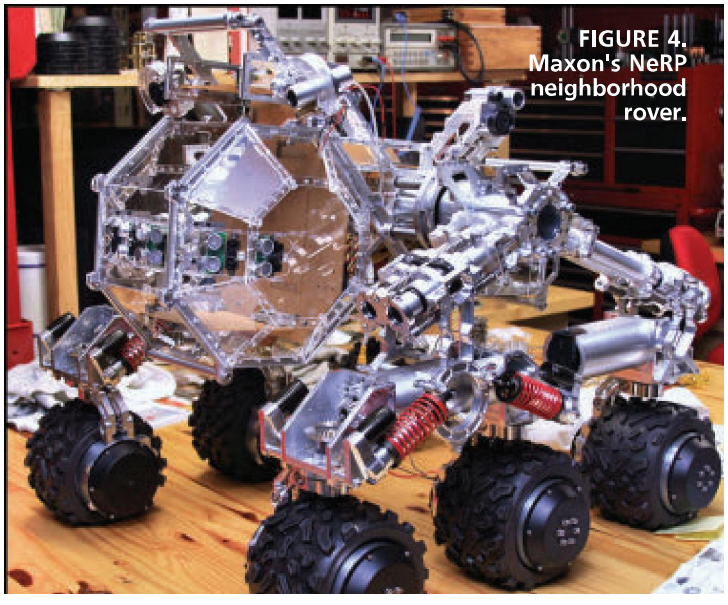


FIGURE 4. Maxon's NeRP neighborhood rover.

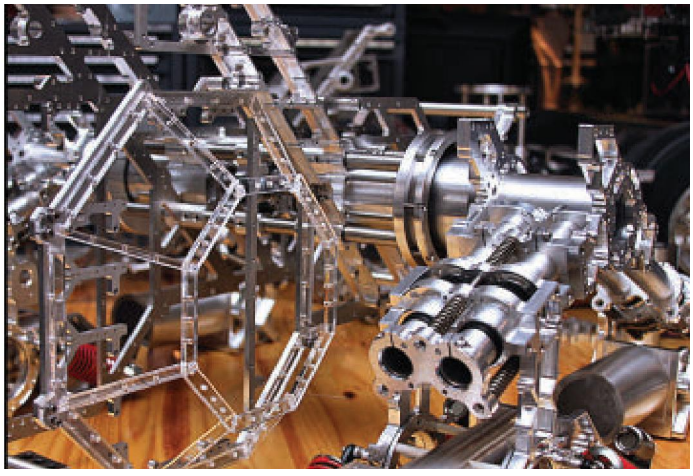


FIGURE 5. NeRP's upper suspension.

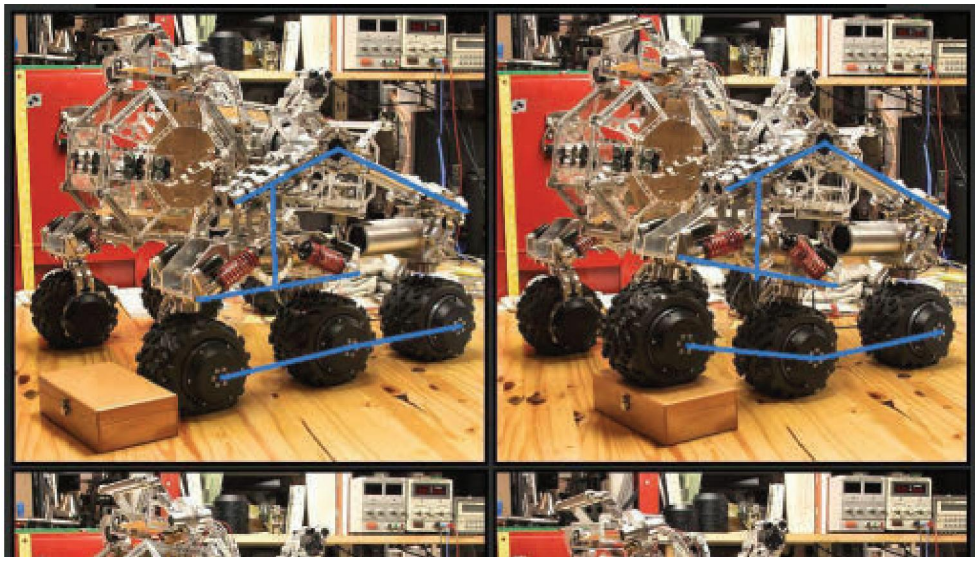


FIGURE 6. NeRP's rocker-bogey wheel system for uneven ground.

FIGURE 7. Part of Ken Maxon's machine shop.

tools that are not visible. Do you need all these machine tools to build a robot like the ones shown in **Figures 3** and **4**? In a word — yes — plus, many more precision metalworking and electronic measurement tools and instruments. More importantly, you need to know good machining techniques, along with electro-mechanical systems knowledge. Do you need these types of tools to build a great robot? No. Dave built Loki with just a small drill press and other hand tools. Be sure and visit Max's Little Robot Shop at **http://nikita.argia.net/kmaxon**.

Robot Design Driven by the Parts That You Find

Designing your robot around parts that you find is the fastest and easiest way to build your robot. Designing a robot on paper or CAD drawings, and then going out and trying to find uniquely shaped parts that exactly or closely match your specs will do nothing but frustrate you. You can certainly machine or fabricate the parts just like Ken did, but a little flexibility can save you a lot of money and time. Plus, you just might end up liking what you find more than a custom-built part.

If you stubbornly insist on 3-53/64" diameter wheels for your robot, you might be ignoring some great wheels at a far cheaper price. With today's microcontrollers and good encoders, you can easily compensate for any wheel's diameter in the program. This is just one example of how flexibility in your design requirements can save you a lot of frustration and money.

Start at the Hardware Store for Robot Parts

Over the years, I have assisted many people in building their first robot. As the years have gone by, I have found more people to be more talented in the use of microcontrollers and associated programming than in the mechanical aspects of robot building. Microcontrollers and a good sensor suite is quite often the basis for a good robot, but mechanics must

come into play for the robot to move at all.

The mounting of the smallest sensor or simple servo drive motors requires some mechanical aptitude. For those with minimal mechanical skills, robot kits such as the LEGO Mindstorms or Parallax BOE-Bot are a great way to go. However, sooner or later, if you don't hang up your robot builder's hat and just give up, you're going to want to design and build your own.

FIGURE 9. Hacking a kid's electric scooter.

FIGURE 8. Aluminum tile transition metal strip.

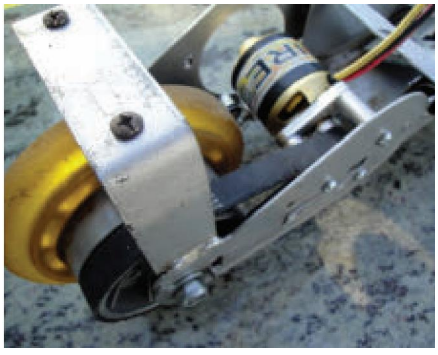


FIGURE 10. A 1,480W scooter motor from Hackaday.com.

As Dave mentioned, you do not have to go to one of the big box stores since your local hardware shop might offer a lot more variety in unique fasteners and bracketry.

Flexible dryer hose makes a very good robot arm covering. Drapery rods and associated hardware are a good source for fastening brackets, as well as pulleys, slides, and metal extrusions. Door hardware, PVC and similar piping in plumbing, plumbing fittings, the hundreds of drawers filled with every fastener imaginable in the hardware section, and unique items in electrical will fill many of your needs.

The Simpson Strong Tie lumber joist hangers and fastening brackets section has steel Ts, Ls, and many other heavy-duty brackets that can assist in holding a robot's frame together. Or, they can actually form a robot chassis.

Plastic sheeting, aluminum, and steel L and H extrusions, steel and brass rod stock, and threaded rod stock can be used to form a complete robot framework. Ls make very good fastening brackets for the long extrusions forming the frame. It is so much easier to bolt angle and square aluminum extrusions together to form a robot's frame than to weld them.

Good welds that result in a truly square framework are difficult as the extrusions many times are tweaked from the heat of the welding process, and disassembly is very difficult if you want to modify things.

One surprising area of a builder's supply is the counter and floor tile area. Tile setters frequently use metal tile transition strips of aluminum shown in **Figure 8**. Made by Schluter and other tile setter accessory companies, these strips have a very nice set of holes that look like the side of a large steel beam used in construction. They come in six foot and longer lengths, and many are L shaped with a 1" to 1-1/2" width. The holes were made to be embedded in the tile's adhesive out of sight, with only the edge showing. These strips make great robot arm structures or to cover the rim around the edges of two robot base metal plates to partially hide the electronics inside.

The power tool section of any hardware store just might have the ideal motors for your next robot. I have seen many small combat type robot builders use cordless drill motors and the associated gearing as drive motors. You certainly do not want to destroy a top-quality drill to rob it of its motor, so go to Harbor Freight or similar stores to locate a basic drill for your project.

I have seen cheap drills with a charger and NiCad battery pack for around $10, but remember, you get what you pay for. You might use the battery pack, but the variable speed control and chuck (as well as the plastic case) will do you no good. A much better bet is checking out garage sales.

Hobby and Craft Shops

Leaving the hardware store, wander over to your local hobby shop. Sure, you will find a lot of R/C equipment and electronic speed controllers, but look at the off-road car parts instead. Knobby tires/wheels with associated shafts, shock absorbers, and mounting hardware have been used for thousands of medium to large robots. Small model airplane wheels make good tires for small robots, but few have any sort of gripping power like the off-road tires, and applying power to them is a bit more difficult.

Motors and drive systems for cars are good but some of the outrunner and brushless motors used in airplanes, helicopters, and quadcopters are not the best for robot motors. Hobby shops often have small bins of square and round tubular brass and aluminum stock that are just great for smaller robots that have intricate mechanical requirements.

Look at the racks of servo linkages, tiny hinges, rods, and special servo horns for great parts for small robots. Craft stores also have parts applicable to robot building such as small tools, wood, plastic and brass plates, adhesives, and all sorts of lacquers and coatings.

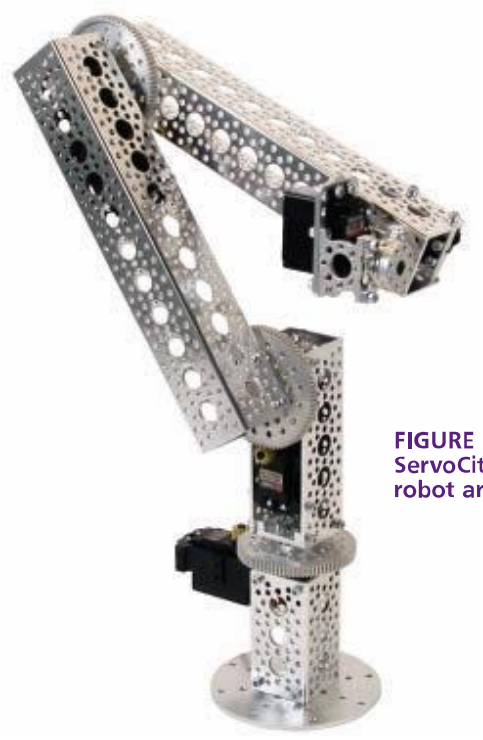


FIGURE 11. ServoCity robot arm.

FIGURE 12. iRobot Create.

FIGURE 13. Yujin Robot Kobuki robot base.

Toy Stores

Toys are unique — especially electrically driven or electronic toys. A toy robot arm might have five axes of motion, whereas an industrial robot with similar capabilities might cost $20,000. The toy industry can accomplish some amazing feats of technology at 0.1% the cost of a similar industrial product. Look at toy cars and toy tanks for the rubber tires and treads that will cost a lot less than products at a robot store. Another area of interest is standup 'Razor' type scooters with electric drives. **Figure 9** shows a simple hacked kid's scooter motor/wheel drive system. Many operate from six or 12 volt SLA batteries that are available everywhere.

Figure 10 shows a souped-up scooter motor from the **hackaday.com** site that runs at 1,480 watts from a 33 volt lithium-ion battery bank. Complete kid's riding cars are great for large-sized robots and they already have the drive system pre-installed. Don't ignore bicycles for the chains, gears, and even smaller tires for robots.

Automobile, RV, and Marine Parts Stores

The best items that I have found in these types of stores are the electric window motors, electric seat motors, and electric door latch actuators. Since cars operate from 12 VDC, most items are ready-made for robots. Car type wiring is usually stranded and far better than solid house wiring for robots. Fuses and fuse holders, watertight RV connectors, and trailer parts are good for robots.

Marine stores also have brass and plastic brackets and mounts that work nicely in robots. Even used (or new) electric trolling motors make great water-borne robot propulsion units, but be careful when using them below a few feet as they are not made for submarines or deep-diving ROVs.

Other Good Sources of Robot Parts

Surplus computer stores have many neat robot parts hidden in old printers. Stepper and DC motors, toothed belts and pulleys, shafts, bearings, rubber rollers, and some good brackets of all types can be found for an hour's worth of disassembly time. Larger office machines and copiers have even bigger robot parts such as chain and sprocket drives, motors, toothed and smooth belts and pulleys, and all sorts of optical items.

Avoid buying many gears unless you have a good way to position the gears within an accuracy of a thousandth of an inch. Use timing belts, instead. Many computer places will gladly give you old printers that they would have had to pay to haul off. Goodwill and thrift stores are also a great source of potential robot parts.

Medical and dental machines can offer a wealth of parts, if you are lucky enough to find some old equipment. Positionable arms to move x-ray heads or other sensors make good robot arms. Rack and pinion drives for exam tables are great for robots. Large syringes — unused — make great one-way hydraulic cylinders for experimental robots.

Security equipment is often used for sensors for robots. PIR sensors, CCD cameras, mounting brackets and enclosures, DC and RF wiring, and even pan and tilt mounts for TV cameras are great for robots. Old style satellite dish or RV trailer linear actuators make

powerful muscles for robots. Appliance repair shops have many moveable items that may need just a simple repair to be used in a robot. Even bent angles of sheet metal from scrap bins can be a mother lode.

Ready-Made Parts for Your Robot

Great platforms and parts designed for robots are available from many manufacturers that advertise here in *SERVO*. Just last month, ServoCity announced a large addition to their robot-specific line of unique parts. I have reviewed many of their products in previous columns, such as the ServoBlocks™ that make for easy adaption of standard model-airplane type servos to the particular needs of robot builders.

Figure 11 shows a sturdy robot arm built using aluminum channel and their new channel-mount servo power gearboxes, ServoBlocks, and hub gears. The use of these linkages, connectors, hubs, adapters, bearings, tubing and shafting, and specialized gears, sprockets, pulleys, and belts can save a builder many hours of searching for parts or just machining them.

It's no secret that I am a real fan of this company and their products. ServoCity has enough unique parts to construct pan-and-tilt platforms and even complete robot arms.

Adding Ready-Made Appendages to a Robot Base

I've got some good feedback from my articles about servos and linear servos and the use of both types of servos in the growing area of robot arms. The larger Invenscience Torxis linear actuators and ServoCity linear servos have been implemented in some great research robots. The smaller and inexpensive Firgelli L12 and L16 miniature linear actuators have been used for robot manipulators on smaller robot bases such as the iRobot Create and larger Parallax bases.

The iRobot Create shown in **Figure 12** with the green command module add-on has proven to be a very popular base for robot experimenters. At less than $130, there are numerous links to programming and constructing robots on this base that has also been used with the Willow Garage TurtleBot.

The Yujin Robot Kobuki experimenter's base shown in **Figure 13** is another platform similar to the Create that Willow Garage has supplied with their newer TurtleBot 2. I am in the process of evaluating a Kobuki that appears to be every bit as capable as the Create.

Parallax developed a large robot base for their Eddie robot but found that their Madeusa base shown in **Figure 14** is also a good seller. It is identical to the Eddie platform but does not have the Kinect and netbook associated hardware. With custom-machined aluminum wheels and pneumatic tires, this is a beauty of a base.

FIGURE 14. Parallax Madeusa robot base.

FIGURE 15. Parallax Stingray robot base.

It is very sturdy and has ample room for a large robot structure. Their lower priced Stingray base with motors shown in **Figure 15** (at less than $100) is another excellent robot base — one that I have used extensively in testing different motor and ESC devices. This 13" x 11" platform has 5" diameter wheels driven by a pair of sturdy 30 watt gearmotors.

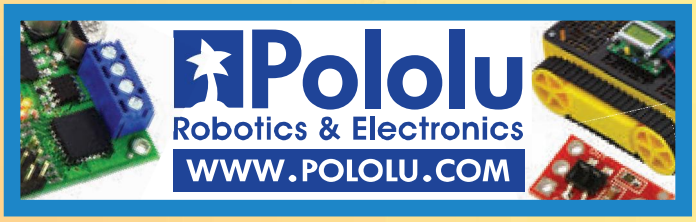
Final Thoughts

Earlier, I talked about two extremes in robot design and construction, as well as how many great robot components and robot platforms are available on the market today. Great robots can be built for just a few dollars or tens of thousands of dollars.

If you really want to build a robot, the sources of parts are unlimited. There are so many groups near most of us that make experienced builders available to help those who are just starting.

Be sure to borrow or buy at least one — if not all — of Gordon McComb's series of *Robot Builder's Bonanza* books, as each one is a great resource. There is absolutely no reason to not get *your* robot up and running. SV

ROBO-LINKS



For the finest in robots, parts, and services, go to www.servomagazine.com and click on Robo-Links.

Advertiser Index

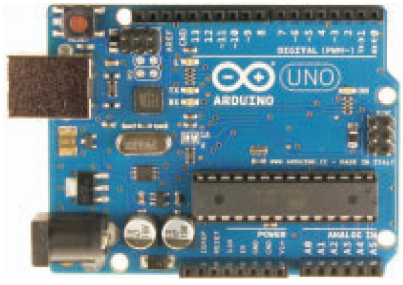
All Electronics Corp. 17, 80
Anaren 66
AndyMark 34, 80
AP Circuits 52
Arizona State University 80
Cana Kit Corp. 80, 81
Command Productions 19
Firgelli 10
HiTec 2, 80
Maker Faire 45
Matrix Robotics Int. 15
Maxbotix 80
PanaVise 52
PCB Pool 34, 80
Pololu Robotics & Electronics ...3, 80
RoboGames 67
Robotis .....Back Cover
RobotShop, Inc 53, 80
Robot Power 44
Servo City/Robot Zone 80, 83
Solutions Cubed 7
SuperBrightLEDs 80
The Robot Marketplace 35
Univ. for Advancing Technology ..82
Vantec 35
Weirdstuff Warehouse 17

CanaKit™
Electronic Kits • Modules • Parts

Save 10% using Coupon Code SM2012* when ordering online at www.canakit.com

Motor Controllers • Arduino • Arduino Kits • Arduino Shields • Robot Platforms • Parts

Arduino Uno



$29.95

ARDUINO-UNO

Latest "R3" Version!

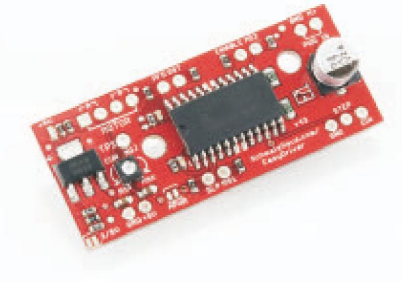
Inventor's Kit for Arduino

ARDUINO-PRK

$94.95

Includes 36-Page Printed Full-Color Manual

EasyDriver Stepper Motor Driver



SX09402

$14.95

Stepper Motor

SX09238

$19.95

50A Motor Speed Controller (PWM)

UK1133

Optional LCD

$59.95

7A Bi-Directional Motor Speed Controller (PWM)

UK1125

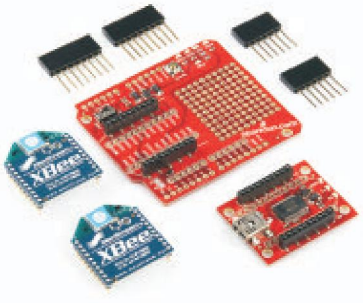
$39.95

USB PIC Programmer (MPLAB Compatible)

UK1300

$69.95

Xbee Wireless Kit



XBEE-KIT

$94.95

1•888•540-KITS (5487) info@canakit.com

www.canakit.com

* Limit one per customer. Can not be combined with other coupons or volume discounts. Does not apply to products already on sale. Prices subject to change without notice.

www.uat.edu/majors

Tech Respect

LEARN

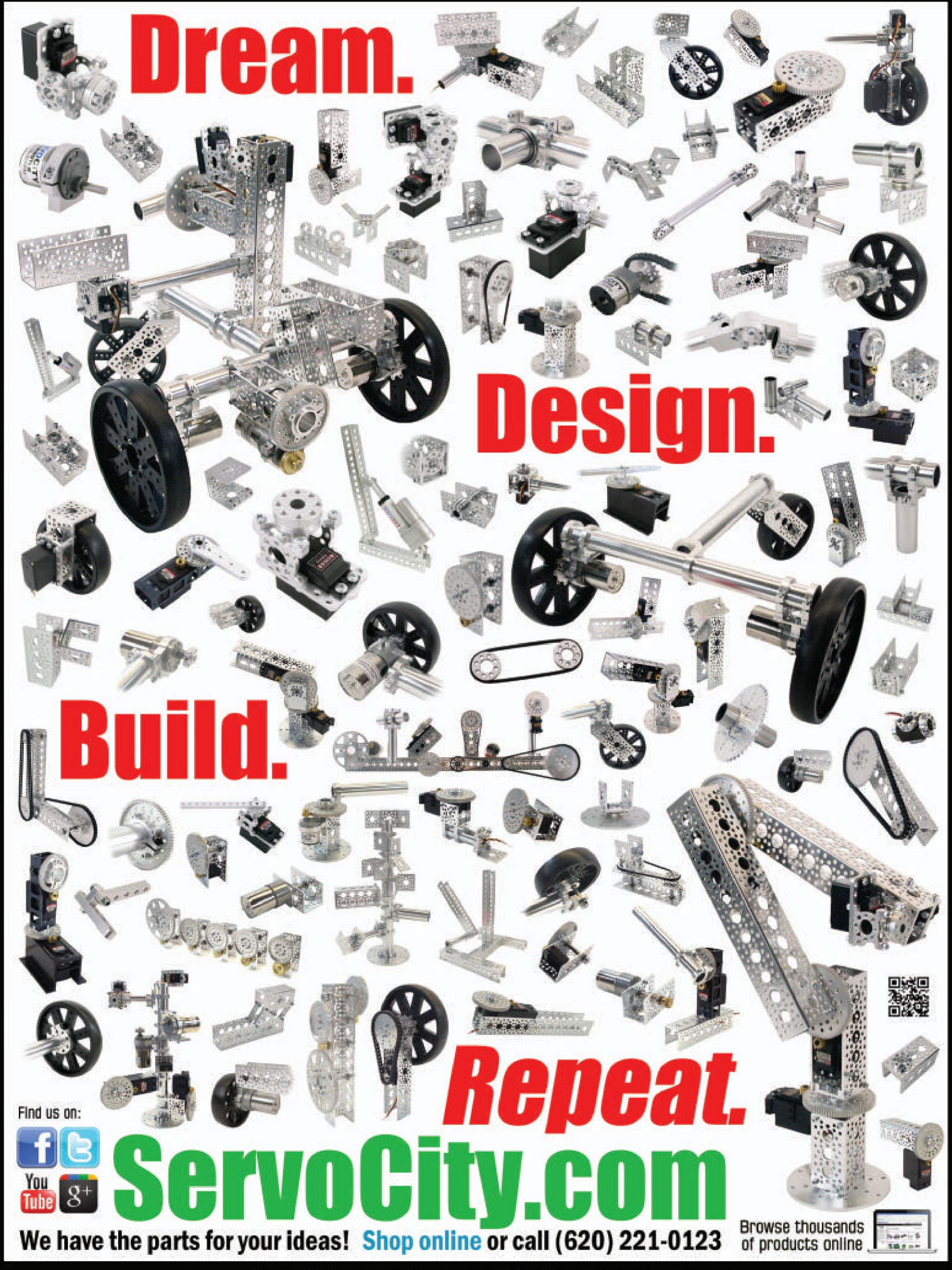
Advancing Computer Science
Artificial Life Programming
Human-Computer Interaction
Open Source Technologies
Robotics and Embedded Systems
Strategic Technology Development
Technology Product Design

Digital Media
Digital Video
Enterprise Software Development
Game Art and Animation
Game Design
Game Programming
Network Engineering

Network Security
Serious Game and Simulation
Technology Forensics
Technology Studies
Virtual Modeling and Design
Web and Social Media Technologies

Bachelors | Masters | Online | On-Campus | www.uat.edu

Please see www.uat.edu/fastfacts for the latest information about degree program performance, placement and costs.



Copyright 1999-2013 Robotzone, LLC · All rights reserved. ServoCity is a registered trademark of Robotzone, LLC.

ROBOTIS

The Actuator Solution for Full Scaled Robots

DYNAMIXEL PRO

All in One-Actuator

High Power, High Precision

Up to 200W BLDC motor and up to 500,000 pulse per turn (4096 absolute resolution)

Full Modular Solution

Ease of implementation (no additional design for parts needed)
Ease of maintenance (simple part swapping)

Sophisticated Control Algorithms

Position and speed input commands with dual-loop current control

Novel Gear Reduction System

High torque output, light weight, with high impact resistance

Item	Unit	Data		
		H42-20-S300-R	H54-100-S500-R	H54-200-S500-R (Preliminary Product)
Rated voltage	V	24	24	24
No load speed	RPM	28.3	35.2	35
No load current	A	0.61	1.06	1.18
Continuous speed	RPM	15.59	32.7	32.1
Continuous torque	Nm	5.596	21.142	39.131
Continuous current	A	1.989	5.930	9.505
Maximum output power	W	23.64	144.58	262.66
Resolution	Step/turn	304,000	502,000	502,000
Gear ratio	-	304	502	502
Backlash	arcmin	3.5	3.5	3.8
Interface	-	RS-485 / CAN	RS-485 / CAN	RS-485 / CAN
Operating temperature	˚C	5~55	5~55	5~55

USA E-mail : america@robotis.com, Tel : +1-949-333-3635
JAPAN E-mail : japan@robotis.com, Tel : +81-3-4330-3660
Etc. E-mail : contactus2@robotis.com, Tel : +82-70-8671-2609
OLLO, BIOLOID, and DYNAMIXEL are registered trademarks of ROBOTIS Co., Ltd.

www.robotis.com